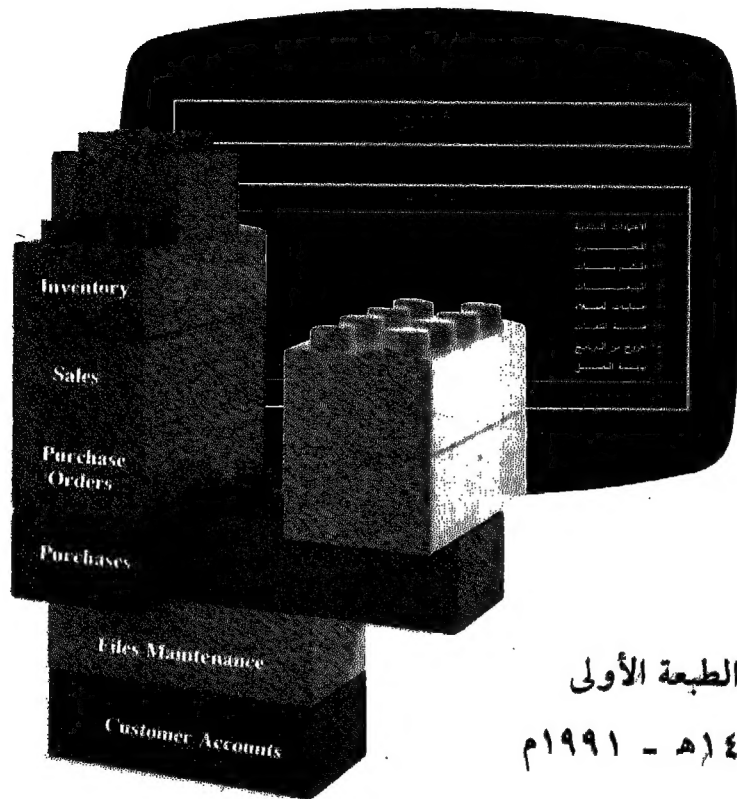


التطبيقات التجارية

باستخدام قاعدة البيانات

dBASE III PLUS



الطبعة الأولى

١٤١٢ هـ - ١٩٩١ م

اهداءات ٢٠٠٢

المهندس/ سيد مصطفى أبو السعود

القاهرة



بجزي محمد كز البر الوطيا

التطبيقات التجارية

باستخدام قاعدة البيانات

dBASE III PLUS

الطبعة الأولى

١٤١٢ هـ - ١٩٩١ م

توزيع مؤسسة جمال الحاسب للإلكترونيات
ص.ب. ١٠٢ البمام ٣١٤١١ - فاكس ٨٣٢٠٤٥١ / ٩٦٦-٣
تلفون ٨٣٢٢١٠٩ / ٨٣٢٢١٤٨

حقوق الطبع محفوظة

حقوق الطبع والنشر محفوظة للمؤلف ولا يجوز نشر أي جزء من هذا الكتاب أو إعادة طبعه أو تصويره أو اقتزاع مادته العلمية بأية صورة دون موافقة كتابية من المؤلف.

أجيز طباعة هذا الكتاب بموجب خطاب سعادة مدير عام المطبوعات بوزارة الإعلام
بالمملكة العربية السعودية رقم ٨٠٧٤/م بتاريخ ٢٧/١٢/١٤١١هـ



المراجع الأساسي

قاعدة البيانات Clipper

يتخاطب هذا الكتاب كلا من مبرجي قاعدة البيانات dBASE III PLUS ومن يرغبون في تطوير نظم لإدارة قواعد البيانات باستخدام

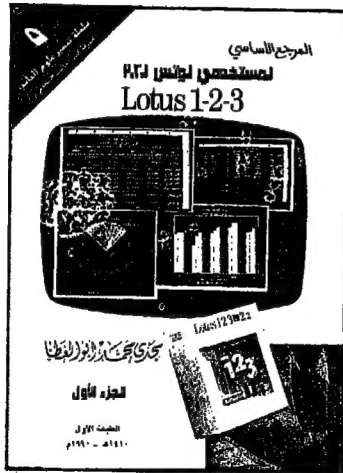
قاعدة البيانات Clipper والكتاب يشتمل على أربعة أبواب على النحو التالي:

الباب الأول: يشرح مفاهيم أساسية عن تاريخ «كلبر» ومتطلباتها وملفاتها وإمكانياتها وضرورة استخدامها في تطوير النظم والفرق بين المفسر والمترجم ويشرح لمبرجي dBASE III PLUS كيفية توفيق برامجهم قبل ترجمتها باستخدام «كلبر» ويركز على الامكانيات التي يتميز بها «كلبر» عن «دي بيس» في تطوير البرامج والنظم.

الباب الثاني: يشرح مفاهيم متقدمة تهتم بصفة أساسية الذين يرغبون في تطوير أنظمة إدارة قواعد البيانات بإمكانيات متقدمة لا توفرها «دي بيس ثري بلاس» مثل المصفوفات واستخدام قوائم الاختيارات ذات الشريط المضاء والتعامل مع شبكات الاتصالات وكيفية التعامل مع أخطاء البرامج وتعقب واكتشاف الأخطاء.

الباب الثالث: يشرح نظاما متكاملا للمبيعات يشتمل على إجراءات وبرامج حية يمكن استخدامها بصورتها الراهنة أو بعد توفيقها لاعداد نظم إدارة قواعد بيانات مشابهة، والنظام يصلح لخدمة مستفيد واحد أو مجموعة مستفيدين داخل شبكة اتصالات محلية.

الباب الرابع: يشتمل على مرجع شامل لجميع الأوامر والوظائف مرتبة ترتيبا أبجديا لسهولة الوصول إلى أي منها، ويشتمل كل أمر أو وظيفة على معلومات وافية تشمل: شرح مختصر، الشكل العام، الاختيارات المتاحة، الشرح، الاختلاف عن «دي بيس ثري بلاس»، مثال على الأقل، الأوامر والوظائف الأخرى ذات الصلة.



المراجع الاساسي

لمستخدمي لوتس ١-٢-٣

Lotus 1-2-3

يشرح هذا الكتاب واحدا من أقوى
البرامج المتكاملة التي تتيح إعداد صفحة
البيانات الالكترونية والرسوم البيانية

وقواعد البيانات. والكتاب يخاطب المبتدئين وأصحاب الخبرة السابقة باستخدام برنامج
LOTUS 1-2-3® ولوتس ١-٢-٣. فبدأ بتقديم نظرة عامة عن برامج صفحة البيانات
الالكترونية وبرنامج LOTUS 1-2-3® بصفة خاصة. ثم تدرج في شرح جميع الامكانيات
الأخرى التي تيسر إعداد صفحة البيانات الالكترونية وإعداد وطباعة الرسوم البيانية وبناء
قواعد البيانات والمختزلات. أما أصحاب الخبرة السابقة باستخدام البرنامج فسيجدون فائدة
عظيمة من خلال التمارين العملية التي يشتمل عليها الكتاب والتي تزيدهم فهما لامكانيات
البرنامج، وتعتبر نماذجاً حية يمكن الاسترشاد بها في حياتهم العملية. فهذه التمارين تشرح جميع
الأوامر اللازمة لاعداد التطبيقات المتكاملة التي تشتمل على صفحة البيانات الالكترونية
والرسوم البيانية وقواعد البيانات والمختزلات (MACROS).

وإنشاما للفائدة تناول الكتاب كيفية استخدام البرنامج مع البيانات العربية باستخدام
جميع برامج تعريب مدخلات ومخرجات الحاسب الموجودة بالساحة حتى الآن.



يشرح هذا الكتاب كيفية استخدام
قاعدة البيانات DBASE III PLUS مع
الحاسبات الشخصية سواء من ناحية
الأوامر واستخدام شاشات المساعدة أو من ناحية البرمجة .

والكتاب صيغ بأسلوب سهل ليخاطب أولئك المشتغلين في مجال الحاسبات ومن
ليست لهم خبرة سابقة بالحاسبات الآلية فقد بدأ بشرح أساسيات واستخدامات الحاسبات
الآلية في الجزء الأول قبل شرح قاعدة البيانات وأوامرها والتعامل معها . كما تم شرح
أساسيات البرمجة في الجزء الثاني منه قبل شرح مفهوم واستخدام البرمجة في قاعدة البيانات .
ولذلك فقد جاء هذا الكتاب بحق مرجعاً أساسياً للمشتغلين والدارسين في هذا
المجال . فقد كتب بأسلوب تعليمي منظم يصلح للتدريس في الجامعات والمعاهد العلمية .
وقد جاء شاملاً أيضاً لكل ما تحتويه المادة ولكل ما يحتاج إليه العاملون في هذا المجال .



يخاطب هذا الكتاب المبتدئين في إعداد نظم إدارة قواعد البيانات باستخدام قاعدة البيانات dBASE IV وذوي الخبرة الطويلة بالأصدارات

السابقة من قاعدة البيانات "dBASE" مثل dBASE III PLUS.

- فالجزء الأول من الكتاب يأخذ بيد القارئ خطوة خطوة من خلال تدريبات عملية مبسطة ليضع بين يديه أساسيات قاعدة البيانات dBASE IV والتي تتلخص في:
- مفهوم قواعد البيانات وتنظيم ملفات ومجالات استخدامها.
- إنشاء الملفات وإدخال بياناتها واستعراض محتوياتها والاستفسار عنها بشتى الطرق.
- ترتيب وتنظيم الملفات وإجراء العمليات الحسابية على بياناتها الرقمية.
- تصميم واستخراج التقارير والملصقات.

ويشرح الجزء الثاني البرمجة باستخدام قاعدة البيانات dBASE IV من خلال مجموعة كبيرة من البرامج معدة بطريقة تعليمية تتدرج من النظرية إلى التطبيق ومن الفهم إلى العمل مع التركيز على المفاهيم الجديدة والتي لم تكن موجودة بالأصدارات السابقة. ويشرح الجزء الثالث مفاهيم متقدمة في قاعدة البيانات تتضمن إعداد نظم شاملة باستخدام مصمم التطبيقات تستخدم التسهيلات التي أضافتها قاعدة البيانات «دي بيس فور». تعتبر نماذج حية يمكن الاقتداء بها لمن يريدون إعداد نظم مماثلة.

وإتماماً للفائدة فقد اشتمل الكتاب على خمسة ملاحق هامة لا يستغني عنها أحد ممن يعملون في هذا المجال.

الحمد لله والصلاة على رسول الله وآله وصحبه ومن والاه . . . وبعد . .
إذا كنت تريد أن تبدأ من حيث انتهى الآخرون فهذا الكتاب من
أجلك. فالكتاب يخاطب المبرمجين الذين يرغبون في تطوير نظم إدارة قواعد
البيانات وبيعها في الأسواق ورجال الأعمال الذين يودون تطوير نظم تجارية
لخدمة أعمالهم.

وهذا هو الكتاب الثاني في سلسلة تطوير نظم المعلومات التي حازت
على ثقة واهتمام القراء العرب، وقد بذل فيه مؤلفه جهداً عظيماً وجاء بالجديد
الذي يسهل فهم واستيعاب التطبيقات المعقدة.
والكتاب يشتمل على التطبيقات التالية:

- نظام الاعتمادات المستندية
- نظام المشتريات
- نظام المبيعات
- نظام المخازن
- نظام حسابات العملاء
- نظام صيانة الملفات

ويمكن تشغيل هذه الأنظمة منفردة أو ربطها معاً كما يمكن تعديلها حسب
رغبة العميل وبيعها في الأسواق.

والله نسأل أن ينفع بهذا الكتاب أبناء العرب والمسلمين وأن يجازي
مؤلفه عنه خيراً.

مصطفى الحسيني

مركز أبحاث شبكة الكمبيوتر الشخصي

الكتاب في سطور

يشتمل كتاب التطبيقات التجارية باستخدام قاعدة البيانات dBASE III PLUS على أحد عشر فصلاً وثلاثة ملاحق على النحو التالي:

الفصل الأول . . . يشرح خطوات تطوير نظم إدارة قواعد البيانات ثم يوضح كيفية تحسين أداء النظم التي يتم تطويرها بصفة خاصة بقاعدة البيانات dBASE III PLUS

الفصل الثاني . . . يشرح مفاهيم هامة مثل تكييف قاعدة البيانات، والتعامل مع حزم برمجية أخرى واستخدام محرر سطور لكتابة البرامج والتعامل مع ذاكرة الحاسب.

الفصل الثالث . . . يشرح فكرة التطبيقات التي يشتمل عليها الكتاب والوظائف التي تؤديها والبرنامج الذي يتولى ربط هذه الأنظمة من خلال قائمة واحدة.

الفصل الرابع . . . يشرح نظام الاعتمادات المستندية والوظائف التي يتطلبها مثل إضافة اعتماد جديد أو تعديل أو إلغاء اعتماد والاستفسار عنه وكيفية استخراج التقارير المطلوبة.

الفصل الخامس . . . يشرح نظام المخازن وكيفية استخراج التقارير اللازمة.

الفصل السادس . . . يشرح نظام المشتريات كجزء من نظام المخازن ويوضح كيفية تسجيل فاتورة الشراء بملف تاريخي للمشتريات وكيفية استخراج التقارير المطلوبة عن فاتورة أو اعتماد أو مشتريات مدة معينة.

الفصل السابع . . . يشرح نظام المبيعات وهو استكمال للفصلين الخامس والسادس ويشرح كيفية تسجيل المبيعات ومردوداتها والاستفسار عنها .

الفصل الثامن . . . يشرح نظام حسابات العملاء ويشتمل النظام على وظائف إضافية أو تعديل أو حذف أو استرجاع حساب عميل بالإضافة إلى تسجيل المديونيات والتسديدات في حسابات العملاء . ويشتمل النظام كذلك على كيفية استخراج التقارير والملصقات وتشمل التقارير تقارير عن العملاء حسب ترتيب معين أو حسب نوعية مختارة منهم وتقارير عن الحسابات مثل حالة حساب معين وكشوف الحسابات الشهرية . وبيان بالحركة المدينة والدائنة للحسابات . وملخص لحسابات العملاء .

الفصل التاسع . . . يقدم هذا الفصل العديد من البرامج المساعدة التي تهتم بصيانة الملفات مثل حفظ واسترجاع الملفات وفهرستها وتنظيفها وتحديث بياناتها بعد عمليات البيع أو الشراء .

الفصل العاشر . . . يشرح كيفية تعريب النظام باستخدام أشهر برامج تعريب مدخلات ومخرجات الحاسب المتاحة .

الفصل الحادي عشر . . . يشرح كيفية تعقب وتصحيح أخطاء البرامج .

الملحق الأول . . . الشفرة الأمريكية لتبادل المعلومات (ASCII) .

الملحق الثاني . . . الحزم البرمجية المتوافقة مع dBASE® .

الملحق الثالث . . . ملف الإجراءات .

المحتويات

١	مقدمة
	الفصل الأول
٥	تصميم نظام إدارة قواعد البيانات
٦	ما هو نظام إدارة قواعد البيانات
٦	خطوات إعداد نظام إدارة قاعدة البيانات
١١	تحسين أداء النظام
١٢	أولاً: استخدام ملف إجراءات
١٤	ثانياً: ترجمة برامج النظام
	الفصل الثاني
١٥	مفاهيم هامة قبل تطوير النظام
١٦	تكييف قاعدة البيانات
٢١	التعامل مع برامج أخرى
٢٢	استخدام محرر سطور لكتابة البرامج
٢٣	التعامل مع ذاكرة الحاسب
	الفصل الثالث
٢٧	إعداد النظام
٢٩	فكرة النظام
٣٠	تحديد الهدف من النظام
٣٢	القائمة الرئيسية للنظام
٣٤	البرنامج الرئيسي
٣٨	استخدام ملف الإجراءات
	الفصل الرابع
٤٥	نظام الاعتمادات المستندية
٤٦	تصميم ملف قاعدة البيانات
٤٨	إنشاء ملفات الفهرسة

٤٩	تحديد برامج النظام
٥١	قائمة نظام الاعتمادات المستندية
٥٣	إضافة اعتماد جديد إلى الملف
٦٥	تعديل بيانات اعتماد
٦٨	حذف بيانات اعتماد
٧١	إظهار بيانات اعتماد
٧٤	تقارير الاعتمادات

الفصل الخامس

٨٣	نظام المخازن
٨٤	تصميم ملف قاعدة البيانات
٨٦	إنشاء ملفات الفهرسة
٨٦	تحديد برامج النظام
٨٨	قائمة نظام المخزون الرئيس
٩٠	إضافة صنف جديد إلى أصناف المخازن
٩٣	تعديل بيانات الأصناف
٩٣	حذف صنف موجود
٩٧	تقارير المخازن
١٠٠	برنامج طباعة صنف واحد
١٠٢	برنامج طباعة الأصناف

الفصل السادس

١٠٩	نظام المشتريات
١١١	تصميم ملفات قاعدة البيانات
١١٣	إنشاء ملفات الفهرسة
١١٤	تحديد برامج النظام
١١٥	قائمة نظام المشتريات
١١٧	تسجيل مشتريات جديدة
١٢٨	الاستفسار عن مشتريات اعتماد
١٣٨	الاستفسار عن مشتريات خلال فترة محددة

الفصل السابع

١٤١	نظام المبيعات ..
١٤٢	تصميم وإنشاء ملفات النظام
١٤٤	إنشاء ملفات الفهرسة
١٤٥	تحديد برامج النظام
١٤٦	قائمة نظام المبيعات
١٤٨	تسجيل فواتير البيع
١٥٨	طباعة فاتورة البيع
١٦١	معالجة مردودات المبيعات
١٦٧	التحقق من كلمة السر
١٦٩	استخراج تقارير البيع

الفصل الثامن

١٧٥	نظام حسابات العملاء ..
١٧٦	تصميم ملفات قاعدة البيانات
١٧٩	إنشاء ملفات الفهرسة
١٨٢	تحديد برامج النظام
١٨٢	قائمة نظام حسابات العملاء
١٨٦	إضافة حساب عميل جديد للملف العملاء
١٨٩	تعديل بيانات حساب عميل
١٩٠	حذف بيانات عميل من الملف
١٩٤	استرجاع سجل معلّم لغرض الحذف
١٩٥	تسجيل العمليات المدينة والدائنة
٢٠٢	تقارير حسابات العملاء
٢٠٦	تقرير بيانات العملاء
٢٠٩	١ - تقرير بيانات العملاء مرتب أبجدياً
٢١٢	٢ - تقرير بيانات العملاء مرتب حسب رقم الحساب
٢١٢	٣ - تقرير بيانات عملاء مختارين
٢١٧	٤ - طباعة الملصقات
٢١٩	الاستفسار عن حالة حساب

٢١٩	تقرير بحركة الحساب المدنية والدائنة
٢٢٦	أولاً: اظهار حركة الحساب على الشاشة
٢٣١	ثانياً: طباعة حركة الحساب على الطباعة
٢٣٤	طباعة ملخص الحسابات
٢٣٨	طباعة كشف الحساب الشهري
	الفصل التاسع
٢٤٩	صيانة الملفات
٢٥٢	قائمة صيانة الملفات
٢٥٢	عمل نسخ احتياطية للملفات
٢٥٥	استرجاع الملفات
٢٥٦	فهرسة الملفات وإعادة فهرستها
٢٦٠	تنظيف الملفات من السجلات غير الضرورية
٢٦١	تعديل أرصدة المخازن بعد عمليات البيع والشراء
	الفصل العاشر
٢٦٧	تعريب النظام
٢٦٨	تعريب النظام باستخدام مساعد العربي/٢
٢٦٨	تشغيل مساعد العربي
٢٦٩	اختيار اتجاه الكتابة على الشاشة واللغة
٢٧٠	تشغيل النظام بالطريقة العربية
٢٧٦	اقتراحات لتيسير كتابة البرامج
٢٧٨	تعريب النظام باستخدام نافذة
٢٧٨	تشغيل نافذة
٢٧٩	اختيار اتجاه الكتابة على الشاشة واللغة
٢٨٠	تشغيل النظام بالطريقة العربية
	الفصل الحادي عشر
٢٨٣	تعقب وتصحيح أخطاء البرامج
٢٨٤	الأخطاء النحوية
٢٨٥	الأخطاء المنطقية
٢٨٦	تغيير حالة بعض الدوال

٢٨٦	إضافة أو امر لتعقب واكتشاف الأخطاء
٢٨٩	الملاحق
٢٩٠	الملحق الأول: الشفرة الأمريكية لتبادل المعلومات (ASCII)
٢٩٥	الملحق الثاني: الحزم البراعجية المتوافقة مع «دي بيس»
٢٩٩	الملحق الثالث: ملف الاجراءات DBPROC.PRG

مقدمة

إن الحمد لله . نحمده ونستعينه ونستعديه ونستغفره ونشكره ونصلي ونسلم على خير خلقه وخاتم رسله محمد بن عبدالله وعلى آله وصحبه أجمعين .

﴿رب اشرح لي صدري . ويسر لي أمري . واحلل عقدة من لساني يفقهوا قولي﴾ . وبعد

أشكر القراء الأعزاء على الثقة التي أولوني إياها بإقبالهم على شراء واقتناء سلسلة **تيسير علوم الحاسب** مما دفعني إلى بذل المزيد من الجهد لتزويد المكتبة العربية بالكتب العلمية المتخصصة فكان هذا الكتاب هو الثامن في هذه السلسلة والثاني من الكتب التي تخاطب الذين استفادوا من سلسلة الكتب التعليمية التي بدأناها مع بداية انتشار الحاسبات الشخصية . والكتاب يشتمل على ستة تطبيقات باستخدام قاعدة البيانات dBASE III PLUS وبالرغم من أننا اخترنا للكتاب اسم التطبيقات التجارية إلا أن البرامج المختارة تصلح لجميع الأغراض وقد قدمنا هذه البرامج بطريقة تعليمية تربوية بحيث تصلح للأغراض التجارية أو لأية أغراض أخرى في نظم إدارة قواعد البيانات .

وقد وضعنا برامج وملفات التطبيقات في صورتها الأولية (المصدرية) مما يسهل لك إمكانية تعديلها أو توفيقها حسب احتياجاتك أو احتياجات العميل لأنك لن تجد من البرامج الجاهزة ما يتواءم مع احتياجاتك الخاصة بالكفاءة المرجوة بل من الممكن ألا تتواءم أنت مع طريقة عمل البرامج الجاهزة الموجودة في السوق والتي يمكنك شراؤها .

وقد اخترنا شرح التطبيقات باستخدام قاعدة البيانات dBASE® لأنها عرفت عبر إصداراتها المختلفة ابتداءً من dBASE II ثم dBASE III ثم dBASE III PLUS وأخيراً dBASE IV بأنها الأداة المرنة والمناسبة جداً للتطبيقات التجارية المعقدة ولغيرها من نظم إدارة قواعد البيانات .

فإذا كنت من المهتمين بإعداد نظم لإدارة قواعد البيانات بعيداً عن قيود وتعقيدات لغات البرمجة التقليدية وإذا كنت تريد أن تبدأ من حيث انتهى الآخرون

- فيجب أن تعرف قاعدة البيانات dBASE[®] وذلك للأسباب الآتية :
- ١ - سهولة استخدامها وتعلمها وذلك لأنها تستخدم الكلمات الإنجليزية اليومية بنفس المعنى أو بمعنى قريب من الموجود في قاموس اللغة على سبيل المثال DISPLAY تعني إظهار و APPEND تعني إضافة في نهاية الملف .
 - ٢ - أن التطورات والتغيرات في التطبيقات التجارية باستخدام قاعدة البيانات dBASE[®] أقل صعوبة من غيرها من لغات البرمجة التقليدية مثل الكوبول أو البيسك أو الفورتران ومن المسلم به أن كل منظمة أو مؤسسة تتطلع دائماً إلى التطور والتوسع والنمو في أعمالها ويتحقق ذلك بإضافة برامج جديدة إلى تطبيقات قاعدة البيانات حتى تلتقي واحتياجاتها المتجددة .
 - ٣ - باستخدام ملفات قاعدة البيانات يتمكن الأفراد ذوي الخبرة المحددة بالحاسب من التعامل مع التطبيقات من حيث الإدخال وإعداد التقارير بالطريقة المألوفة .

لمن هذا الكتاب

هذا الكتاب أعد من أجل مستخدمي الحاسبات الشخصية والمهتمين بها من مديريين ومشرفين ومبرمجين ورجال أعمال وغيرهم من مستخدمي خدمات الحاسب الآلي لتسهيل أعمالهم . وأيضاً فإن هذا الكتاب مادة خصبة لمعاهد ومراكز تعليم الحاسبات سواء للمدرسين أو الدارسين .

والكتاب لا يتطلب سابق خبرة بالبرمجة أو تحليل الأنظمة بيد أنه يتطلب أن يكون لديك خبرة سابقة بقاعدة البيانات dBASE III PLUS . وتريد أن تعرف كيفية عمل برامج تطبيقية باستخدامها . فإذا لم تكن لديك خبرة سابقة بقاعدة البيانات dBASE[®] أو كنت في حاجة لشرح تفصيلي لأساسيات البرمجة باستخدامها فيجب أن تقرأ أولاً كتابنا المرجع الأساسي لقاعدة البيانات dBASE III PLUS الذي يشتمل على شرح تفصيلي ومبسط لاستخدام قاعدة البيانات في الجزء الأول وعلى أساسيات البرمجة باستخدام قاعدة البيانات في الجزء الثاني .

محتويات الكتاب

يضع هذا الكتاب بين يديك البداية السليمة لتطوير نظم إدارة قواعد البيانات وتحقيق احتياجاتك واحتياجات الآخرين من خلال ستة تطبيقات بالإضافة إلى احتوائه على نماذج يحتذى بها في تطوير برامجك الخاصة. كما يشتمل على برامج عامة تصلح لأغراض صيانة الملفات مهما كان نوع النظام المستخدم. ويشتمل على فصل خاص لتعريب التطبيقات باستخدام أشهر برامج تعريب مدخلات ومخرجات الحاسب وهي نافذة ومساعد العربي / ٢ وفصل آخر لشرح كيفية تعقب واكتشاف الأخطاء. وقد جاء هذا الكتاب شاملاً لأنني قدمت فيه كثيراً من المفاهيم الضرورية للعاملين في هذا المجال مثل مفاهيم البرمجة التركيبية وتحليل الأنظمة في حين تهتم الكتب الأخرى بشرح المادة شرحاً جامداً معتمدة على خرة القارئ في منطق البرمجة وتحليل الأنظمة وهذا ما لا يروق للمبتدئين في هذا المجال أو الذين يريدون أن يبدأون من حيث انتهى الآخرون.

وقد تم اختيار جميع البرامج والأنظمة الواردة بهذا الكتاب بعناية ودقة فائقتين. ولذلك فقد جاءت هذه الأنظمة نماذج متكاملة تشتمل على الوظائف الأساسية التي لا يستغنى عنها أي نظام لإدارة قواعد البيانات يمكن أن تنفذ كما هي أو بعد إجراء تعديلات بسيطة عليها لتوافق حاجتك الخاصة. ومن خلال البرامج التي تم شرحها في هذا الكتاب تعرضنا لجميع الأوامر والوظائف التي يجدر الاهتمام بها.

وبعد.. . فإنني أقدم شكري لله عز وجل أن وفقني لإتمام هذا العمل كما أشكر القراء الأعزاء على الثقة التي أولوني إياها بإقبالهم على شراء واقتناء **سلسلة تيسير علوم الحاسب** التي نرجو من الله أن تكون لبنة في خدمة الإسلام والعرب جميعاً.

«وآخر دعوانا أن الحمد لله رب العالمين».

مجدي محمد أبو العطا

الفصل الأول

تصميم نظم إدارة قواعد البيانات

يسرّج هذا الفصل خطوات تطوير نظم إدارة قواعد البيانات بصفة عامة. سواء كانت هذه النظم سيتم تطويرها باستخدام قاعدة البيانات dBASE III PLUS موضوع دراسة هذا الكتاب أو باستخدام لغة أخرى من لغات البرمجة المعروفة. ثم يوضح كيفية تصميم أداء النظم التي يتم تطويرها بصفة خاصة باستخدام قاعدة البيانات dBASE III PLUS ويقترح طريقتين:

• استخدام ملف إجراءات (Procedure file)

• ترجمة برامج النظام (Compiling)

ما هو نظام إدارة قواعد البيانات

يتكون نظام إدارة قاعدة البيانات من مجموعة من الملفات بالإضافة إلى البرنامج أو مجموعة البرامج التي تتضافر معًا لحل مشكلة أو لتحويل نظام يدوي إلى نظام يعمل بالحاسب، مثل تحويل نظام حسابات العملاء أو حسابات الموردين من نظام ودفاتر يدوية إلى نظام وملفات تستخدم بواسطة الحاسب ويمكن أن يشتمل النظام على برنامج واحد وفي هذه الحالة فإن هذا البرنامج مع ملفات النظام يطلق عليه نظام إدارة قاعدة البيانات أو قد يشتمل على مجموعة من البرامج - بالإضافة إلى ملفات النظام - وفي هذه الحالة فإن هذه البرامج مجتمعة يطلق عليها نظام إدارة قاعدة البيانات.

وستناول فيما يلي خطوات التصميم الجيد لنظام إدارة قواعد البيانات ثم نورد اقتراحات هامة لتحسين أداء النظم التي يتم تطويرها باستخدام قاعدة البيانات dBASE III PLUS.

خطوات إعداد نظام إدارة قاعدة البيانات

يجب أن يسبق كتابة برامج نظام إدارة قاعدة البيانات تصميم جيد لهذا النظام لأن التصميم الجيد للنظام لا يقل أهمية عن كتابة البرامج أو تنفيذها وفي النظم الصغيرة التي تقتصر مهمتها على وظيفة واحدة مثل إعداد شاشات الإدخال للنظام أو استخراج تقرير يمكنك كتابة البرنامج اللازم للنظام مباشرة بدون حاجة لإعداد تصميم سابق لهذا النظام. أما في النظم الكبيرة والمعقدة فلا بد أن تضع خريطة للنظام توضح البرامج التي يشتمل عليها هذا النظام وعلاقة كل برنامج بالآخر. وفيما يلي سنوضح خطوات التصميم الجيد لنظام إدارة قواعد البيانات.

① تحديد الهدف من النظام

يجب أن يكون الهدف من النظام أو وظيفة النظام واضحة تمامًا في ذهن الشخص الذي سيقوم بإعداد هذا النظام وتتطلب هذه المرحلة وجود شخص له دراية فنية بالنظام

أو الجلوس مع شخص ذو دراية فنية بالنظام . فمثلاً لإعداد نظام محاسبة يجب أن يعده محاسب أو أن يجلس أخصائي الحاسب أو المبرمج مع المحاسب لشرح له طريقة العمل اليدوية والمستندات المستخدمة والدورة المستندية التي تمر بها ليستطيع أخصائي الحاسب أن يترجم هذه المستندات والدورة المستندية التي تمر بها إلى مستندات ودورة مستندية مماثلة باستخدام قاعدة البيانات فإذا أصبحت أهداف النظام واضحة فيجب الانتقال إلى المرحلة التالية .

٢) تحديد متطلبات النظام

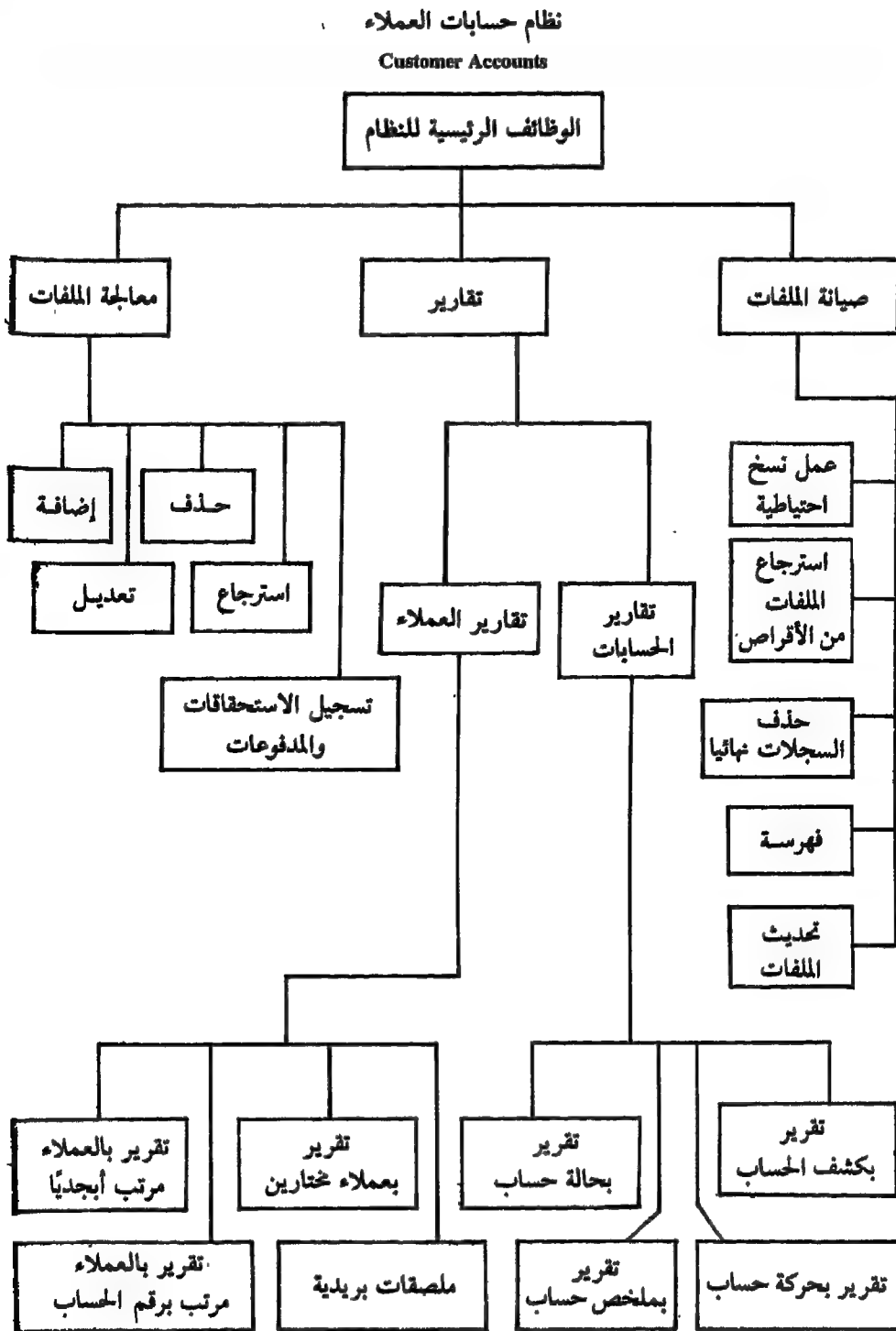
وفي هذه الخطوة يتم تحديد الملفات المستخدمة في النظام بدقة ووضوح مثل أسماء الملفات والحقول التي تشتمل عليها وطرق ترتيبها والبحث فيها والتقارير المستخدمة وأشكالها وشكل شاشات الإدخال ومحتوياتها . . . إلخ . كما يجب تحديد طريقة الحصول على الخدمات التي يتطلبها النظام مثل نسخ الملفات على أقراص تحفظاً للتلف أو الفقدان، أو استرجاع محتويات الملفات عند الحاجة إليها ويتم أيضاً اختيار شكل القوائم الرئيسية والفرعية التي يشتمل عليها النظام .

٣) وضع خريطة للنظام

بعد ذلك يجب إعداد خريطة للنظام تشتمل على الوظائف الأساسية المطلوبة من النظام . بحيث يمكن الرجوع إليها لتحديد وظائف النظام ويشتمل شكل ١ - ١ على خريطة نظام حسابات العملاء ومنه تلاحظ أننا قسمنا النظام إلى وظائف والوظائف إلى وظائف فرعية ويتم تقسيم النظام إلى وظائف أساسية والوظائف إلى وظائف فرعية تبعاً للتحليل الأولي الذي أعدته للنظام . وتعتمد هذه الخريطة على مفهوم يسمى التصميم من أعلى لأسفل وهو المفهوم الذي اخترناه لشرح التطبيقات التي يشتمل عليها الكتاب .

٤) إعداد النظام

وتشتمل هذه الخطوة على عدة مراحل نوجزها فيما يلي :
- تصميم ملفات قاعدة البيانات مثل أسماء الملفات ومعرفة الحقول التي سيشتمل



شكل ١ - ١ خريطة نظام حسابات العملاء

- عليها كل ملف .
 - تصميم شاشات إدخال البيانات وتعديلها ومعرفة الحقول التي ستشتمل عليها وموقع كل حقل داخل الشاشة والشكل المقترح للشاشة .
 - تصميم ملف أو ملفات التقارير ومعرفة الأقسام التي سيشتمل عليها التقرير والبيانات المستخرجة من الملفات ومواقعها .
 - تصميم ملفات الملصقات ومعرفة البيانات التي ستشتمل عليها .
 - تصميم شكل القوائم والاختيارات التي ستشتمل عليها كل قائمة وربط القوائم التابعة بالقائمة الرئيسية وتحديد الإجراء المناسب لكل اختيار من اختيارات القوائم .
- ولأن هذه المراحل هي موضوع هذا الكتاب فسوف نشرحها بالتفصيل في الفصول التالية .

⑤ كتابة برامج النظام

بعد الانتهاء من وضع خريطة النظام وتصميم الملفات والقوائم والشاشات التي يشتمل عليها النظام يجب أن تبدأ في كتابة البرامج التي تتيح الحصول على الشاشات والقوائم والمخرجات التي انتهت من تصميمها . ويجب أن تعرف ما هو الترتيب الذي ستبدأ به في كتابة برامج النظام - وما هو البرنامج الذي ستبدأ به أولاً؟ وهناك طريقتين لهذا الغرض .

إحدى هاتين الطريقتين أن تبدأ بالبرنامج الرئيسي للنظام ثم البرامج الفرعية . . . وهكذا . على منوال الطريقة المتبعة في خريطة النظام شكل ١ - ١ فيكون أول برنامج في النظام هو البرنامج الذي يستدعي القائمة الرئيسية للنظام . وفي هذه الحالة يتم اختبار كل برنامج بعد الانتهاء من كتابته مع مراعاة وضع برامج أو أوامر وهمية محل البرامج التي يستدعيها البرنامج محل الإختبار لتفادي الحصول على رسائل الخطأ في مرحلة التجارب .

الطريقة الثانية أن تبدأ من أسفل الخريطة متجهًا إلى أعلى فتبدأ بالبرنامج أو البرامج التي لا تعتمد على برامج أخرى ثم تختبر البرنامج الذي تنتهي منه - وبعد ذلك تنتقل إلى البرنامج الذي يستدعي البرنامج الذي كتبته وجربته . وهكذا . بحيث يكون آخر برنامج هو البرنامج الذي يستدعي القائمة الرئيسية للنظام .

٦) اختبار النظام

بعد الانتهاء من إعداد النظام واختبار كل مرحلة من مراحله على انفراد يجب اختبار النظام كله . وفي هذه المرحلة يجب إدخال بيانات اختبارية يتم تجربة النظام عليها . ويجب أن يستخدم النظام بجميع الاحتمالات الواردة لاستخدامه من قبل المستخدم النهائي . ولذلك فيجب أن تجلس أمام النظام وتقضي وقتًا طويلاً في اختباره حتى تتأكد أن النظام خاليًا من الأخطاء ومن المفيد أن تدخل بيانات خاطئة لتتأكد أن النظام يكتشفها . وبعد تجربة كل حالة من حالات استخدام النظام راجع بيانات الملفات لتتأكد أن السجل المطلوب إدخاله أو تعديله أو حذفه تم إدخاله أو تعديله أو حذفه فعلاً بمعنى آخر ضع نفسك مكان المستخدم النهائي للنظام الذي ليست لديه دراية عن النظام وفكر بنفس الطريقة التي يفكر بها لأن الشخص الذي ليست لديه دراية عن النظام يحاول تنفيذ الأشياء التي لم ترد بحساباتك وبالتالي فإنه يكتشف الأخطاء التي قد لا تكتشفها أنت .

٧) توثيق النظام

والتوثيق المقصود ذو شقين :

* الشق الأول: توثيق خطوات الحل السابقة وتشمل : أهداف النظام وخريطة النظام ومحتويات البرامج التي يشتمل عليها النظام . ويجب أن توضع كل هذه المستندات داخل ملف خاص بالنظام ليتمكن الرجوع إليه في المستقبل عندما تحتاج لتعديل النظام لأنك إذا حاولت الرجوع إلى نظام أعدته منذ عام مثلاً فستجد أنك نسيت كثيراً من التفاصيل الضرورية . ومن مزايا هذا التوثيق تسهيل عمل أي شخص آخر غير الشخص الذي قام بإعداد النظام إذا أراد إجراء تعديل أو الرجوع إلى أحد البرامج .

* الشق الثاني: إعداد كتيب صغير يشتمل على أهداف النظام بالإضافة إلى التعليمات الضرورية لتشغيل النظام ويجب أن يكون الكتيب مختصرًا وواضحًا بقدر الإمكان. كما يفضل أن يشتمل على أمثلة إرشادية للمستخدم في الحالات التي تحتاج لذلك. وتتوقف طريقة إعداد كتيب التشغيل على مستوى المستخدم النهائي الذي سيتولى تشغيله فكلما كان المستخدم بعيدًا عن النظام كلما احتاج لتوضيح أكثر.

تصميم أداء النظام

بعد الانتهاء من تطوير النظام واختباره ستجد لديك العديد من البرامج والبرامج الصغيرة التي تتضافر معًا لتنفيذ النظام وتشغيله.

وتعتمد فكرة البرمجة التركيبية التي تستخدمها «دي بيس ثري بلاس» على تحميل البرنامج الذي يتم استدعائه من القرص إلى الذاكرة. ولما كانت هذه العملية تستغرق بعض الوقت فيمكنك حسب رغبتك تجميع هذه البرامج في ملف واحد أو أكثر. هذا الملف يطلق عليه Procedure file أو ملف الإجراءات وذلك لزيادة سرعة تنفيذ النظام.

كما يمكنك أيضًا - حسب رغبتك - زيادة سرعة تنفيذ النظام بترجمة (Compiling) هذه البرامج من الصورة المصدرية التي يتم كتابتها بها باستخدام أحد برامج الخدمات التي تأتي ضمن حزمة «دي بيس ثري بلاس». هذا البرنامج اسمه LINKER. إلا أن استخدام هذه الطريقة يتطلب تشغيل البرامج بعد ترجمتها من خلال «دي بيس ثري بلاس». فإذا أردت ترجمة البرامج إلى برنامج جاهز للتنفيذ يطلق عليه Executable file ويخصص له الاسم الممتد .EXE. إذا أردت ترجمة برامجك بهذه الطريقة وتشغيلها مباشرة من محث نظام التشغيل DOS بدون تشغيل «دي بيس ثري بلاس» فنصحك بالرجوع إلى كتابنا الرابع مع الأسامي لقاعدة البيانات Clipper، وهو يوضح كيفية ترجمة برامجك التي طورتها بقاعدة البيانات «دي بيس ثري بلاس» باستخدام Clipper بالإضافة إلى شرح التسهيلات التي تنفرد بها «كلبر». وسنوضح فيما يلي كيفية تحسين أداء النظام بطريقتين:

- استخدام ملف إجراءات
- ترجمة برامج النظام

أولاً: استخدام ملف إجراءات (Procedure file)

في التطبيقات الكبيرة والمعقدة يحتاج الأمر لإعداد عشرات البرامج . وكل برنامج يؤدي وظيفة محددة . وبالرغم من أن تقسيم النظام إلى برامج صغيرة كثيرة يقوم كل منها بوظيفة محددة يسهل إعداد النظام وصيانته فيها بعد إلا أنه يسبب بطء في التنفيذ فكما هو معروف أن استدعاء البرنامج للتنفيذ يتم بإصدار أمر DO الذي يتسبب في نقل البرنامج المطلوب للتنفيذ من وحدة القرص الثابت أو المرن إلى ذاكرة الحاسب ليتم التعامل معه . وبالطبع تُسبب هذه العملية بعض التأخير.

هذا من ناحية ، ومن ناحية أخرى قد نحتاج لإعداد عدد كبير من البرامج الصغيرة وكل برنامج يؤدي وظيفة محددة لا تزيد على بضعة أوامر . فإذا تم وضع كل برنامج داخل ملف مستقل (.PRG) فإن ذلك سيُشغل مساحة كبيرة من وحدة القرص الممغنط وبالطبع فإن ذلك يعتبر ضياعاً لمساحة كبيرة من القرص غير مستخدمة .

فهل من طريقة تزيد سرعة التنفيذ وتوفر مساحة القرص الممغنط التي تستهلكها البرامج الصغيرة . . . ؟

نعم . . . الحل في استخدام ملف يسمى Procedure file ولكن ما معنى كلمة Procedure ؟ . كلمة Procedure في القاموس معناها إجراء ولكنها تعني في قاعدة البيانات مجموعة من التعليقات موجودة داخل برنامج يتم استدعائها لأداء وظيفة محددة ، أو لتنفيذ إجراء معين فعادةً يشتمل ملف الإجراءات (Procedure file) على أكثر من برنامج واحد . ونلجأ إلى هذا النوع من الملفات للتغلب على مشكلة استنفاد مساحة القرص الممغنط التي تسببها البرامج الفرعية الصغيرة . ولاختصار وقت تنفيذ هذه البرامج لأنها توضع في ذاكرة الحاسب ويتم تنفيذ أحدها حال استدعائه مباشرة .

ويتم تحميل البرامج الموجودة بملف الإجراءات في الذاكرة بمجرد فتح الملف باستخدام أمر `SET PROCEDURE TO <file>`

ونوضح فيما يلي كيفية إنشاء واستخدام ملف الإجراءات .

إنشاء ملف الإجراءات واستخدامه

توضع الإجراءات داخل ملف واحد (PRG). وتعرف بداية كل إجراء بأمر `<procedure name> PROCEDURE` ونهايته بأمر `RETURN` بهذا الشكل:

* Program: AAA

PROCEDURE XXX

<Commands>

RETURN

PROCEDURE YYY

<Commands>

RETURN

PROCEDURE ZZZ

<Commands>

RETURN

ولكي تستدعي أحد هذه الإجراءات للتنفيذ يجب أولاً أن تفتح ملف الإجراءات بأمر `<Procedure name> DO` ويمكن أن يستدعي الإجراء الموجود داخل ملف الإجراءات إجراءً آخر داخل نفس الملف. ولكنه لا يستطيع استدعاء إجراء موجود في ملف إجراءات آخر على نفس الدليل أو القرص.

أما إذا أردت إغلاق ملف الإجراءات فيجب استخدام أمر `SET PROCEDURE TO` أو أمر `CLOSE PROCEDURE`

[ملاحظة: راجع ملف الإجراءات المستخدمة في هذا الكتاب (في الملحق الثالث)].

أمور يجب مراعاتها عند استخدام ملف الاجراءات:

- ١ - يجب أن يتناسب حجم ملف الإجراءات مع حجم الذاكرة المتاحة.
- ٢ - يجب أن يتكون ملف الإجراءات من أكثر من برنامج فإذا كان يتكون من برنامج واحد فيحسن وضعه داخل البرنامج الرئيسي أو استدعائه كبرنامج فرعي.
- ٣ - أقصى عدد من البرامج مسموح به داخل ملف الإجراءات هو ٣٢ برنامجاً.

- ٤ - يسمح بفتح ملف إجراءات واحد في نفس الوقت داخل ذاكرة الحاسب بأمر SET PROCEDURE إلا أن النظام الواحد قد يشتمل على أكثر من ملف إجراءات . وفي هذه الحالة يلزم إغلاق الأول قبل فتح الثاني .
- ٥ - ينصح بإعداد ملف الإجراءات بعد الانتهاء من كتابة النظام واختباره وذلك للأسباب الآتية :
- إذا أردت تصحيح خطأ في أحد الإجراءات فلا بد من إغلاق ملف الإجراءات أولاً وإعادة فتحه بعد عمل التصحيحات المطلوبة لكي تتعرف عليها «دي بيس ثري بلاس»
 - أن ملف الإجراءات قد يكون طويلاً وبالتالي يصعب عليك البحث عن إجراء موجود ضمن ٢٠ أو ٣٠ إجراء في نفس الملف .

ثانياً: ترجمة البرامج النظام

ويمكن إجراء ذلك بإحدى طريقتين :

الأولى : استخدام برامج dBCODE

يشرح فصل RUNTIME+ في الجزء الثاني من كتاب الشركة المنتجة لقاعدة البيانات كيفية استخدام هذا البرنامج لترجمة البرامج بعد الانتهاء من تطويرها واختبارها . وهذا البرنامج يزيد سرعة تنفيذ النظام إلا أنك لا تستطيع تعديل البرامج بعد ترجمتها باستخدام dBCODE . ولذلك يلزمك الاحتفاظ بالنسخة الأصلية لإجراء التعديلات عليها وإعادة ترجمتها إذا لزم الأمر ويعيب هذه الطريقة أنها تعتمد على وجود قاعدة البيانات في الذاكرة بالإضافة إلى البرنامج المطلوب للتنفيذ .

الثانية : استخدام مترجم البرامج «دي بيس ثري بلاس»

وفي هذه الطريقة تستخدم أحد البرامج التي تترجم برامج «دي بيس ثري بلاس» وهي كثيرة الآن منها على سبيل المثال Force III _Quicke server _Clipper . ونصح بمراجعة كتابنا المرجع الأساسي للقاعدة البيانات Clipper لمزيد من المعلومات عن ترجمة وربط البرامج وكيفية ترجمتها وربطها .

الفصل الثاني

مفاهيم هامة قبل تطوير النظام

سنعرض في هذا الفصل لمفاهيم هامة تعمل عليك تطوير نظم إدارة البيانات وهذه المفاهيم هي:

- تكييف قاعدة البيانات
- التعامل مع برامج أخرى
- استخدام محرر السطور لكتابة البرامج
- التعامل مع ذاكرة الحاسب

فإذا كانت هذه المفاهيم معروفة لك فيمكنك الانتقال إلى الفصل التالي.

تكييف قاعدة البيانات Customizing dBASE III PLUS

تم تجهيز قاعدة البيانات بقيم تلقائية من قبل الشركة المنتجة. هذه القيم تنظم جو العمل الذي سيتم باستخدام قاعدة البيانات وتسمى هذه القيم التلقائية التي تم تجهيز قاعدة البيانات لتعمل تبعاً لها Default Setting. وبالرغم من أن هذا التجهيز المسبق يوافق معظم المستخدمين إلا أنه قد لا يروق لكثير منهم. وذلك لاختلاف الحاسبات التي يستخدمونها أو لاختلاف أمزجتهم أو طبيعة عملهم.

ويتم تكييف قاعدة البيانات بتغيير بعض القيم الخاصة بنظام التشغيل DOS أو تلك الخاصة بقاعدة البيانات.

ولذلك فإن تكييف قاعدة البيانات هنا يقصد به تهيئة الحاسب من جهة. وذلك بتعديل ملف CONFIG.SYS. وتهيئة قاعدة البيانات من جهة أخرى بتعديل ملف CONFIG.DB وسوف نتناول بالشرح فيما يلي كل من هذين الموضوعين مع إعطاء مثال لكل منهما لتسترشد به عندما تريد تكييف كل من حاسبك وقاعدة البيانات للعمل.

الموضوع الأول: تهيئة الحاسب Configuring the system

يشتمل ملف CONFIG.SYS على التعليمات المستخدمة في تهيئة الحاسب. ففي كل مرة تدير مفتاح تشغيل الحاسب يبدأ نظام التشغيل DOS في البحث في الدليل الرئيسي في الاسطوانة التي بدأت تشغيل الحاسب منها عن ملف اسمه CONFIG.SYS ، فإذا وجد الحاسب هذا الملف فإنه يقرأه ويبدأ في تنفيذ التعليمات الموجودة بداخله . أما إذا كان هذا الملف غير موجود فإن نظام التشغيل يخصص قيماً تلقائية للأوامر التي تهيء الحاسب في بداية العمل.

وعادة تسمح حاسبات آ. ب. م. والمتوافقة معها بحد أقصى قدره ٨ ملفات لتفتح معاً في وقت واحد ويحتاج تشغيل نظام التشغيل إلى خمسة منها. معنى ذلك أن عدد الملفات التي تبقى لتفتح معاً مع قاعدة البيانات هو ٨ - ٥ = ٣ ملفات.

وتسمح قاعدة البيانات بفتح عدد من الملفات بحد أقصى قدره ١٥ ملفاً في نفس الوقت. فإذا حاولت تشغيل قاعدة البيانات بدون إنشاء ملف CONFIG.SYS فإن الحاسب سيخصص لك العدد التلقائي وهو ٨ ملفات منها خمسة لنظام التشغيل ولذلك فإن العدد الباقي وهو ثلاثة ملفات لن يكون كافياً وبالتالي ستحصل على رسالة مفادها أن عدد الملفات المفتوحة تجاوز العدد المسموح به. هذه الرسالة هي:

Too many files are open.

لذلك فيجب أن تنشئ ملفاً اسمه CONFIG.SYS لتغير العدد التلقائي للملفات الذي يخصص بواسطة الحاسب. ويجب أن تلاحظ أنك إذا أنشأت أو عدلت في بعض أوامر ملف التهيئة هذا فإن هذه التعديلات لن يتأثر بها الحاسب ولن ينفذها إلا في المرة القادمة التي تبدأ فيها تشغيل الحاسب.

وقبل إنشاء الملف يجب أن تتأكد أن هذا الملف بهذا الاسم غير موجود من قبل. فإذا كنت تستخدم قرصاً ليئاً (Floppy disk) لنظام التشغيل تأكد أن الملف غير موجود على هذا القرص. وإذا كنت تستخدم قرصاً صلباً (Hard Disk) تأكد أن هذا الملف غير موجود على الدليل الرئيسي (Root Directory) ولكي تتأكد أن هذا الملف غير موجود استخدم الأمر التالي وأنت مع نظام التشغيل DOS:

A> DIR CONFIG.SYS

(وفي حالة استخدام القرص الصلب فإن الموجه (Prompt) سيكون C> بدلاً من A>) وبالطبع ستواجه بإحدى حالتين:

١ - أن الملف موجود من قبل على الدليل.

٢ - أن الملف غير موجود على الدليل.

الحالة الأولى: إذا كان الملف موجوداً من قبل فيجب أن تضيف الأمرين التاليين في نهاية الملف وذلك باستخدام أي منسق للكلمات أو حتى الأمر EDLIN المستخدم مع نظام التشغيل:

FILES = 20

BUFFERS = 15

حيث إن الأمر الأول يخصص عددًا من الملفات مقداره عشرون لتفتح معًا. والأمر الثاني يخصص عددًا من المحطات أو القنوات التي يحجزها نظام التشغيل في ذاكرة الحاسب مقداره خمسة عشر. وقد أخذت هذا الرقم بالذات (١٥) لأنه هو الرقم الذي نصحت به الشركة التي صممت dBASE وعادة يتحدد إجمالي عدد المحطات (Buffers)^(١) التي يمكن تخصيصها في ذاكرة الحاسب بحجم الذاكرة المتاحة في الحاسب المستخدم.

الحالة الثانية: إذا كان الملف غير موجود يجب أن تنشئ ملفًا باسم CONFIG.SYS وتضع بداخله الأمرين المشار إليهما على القرص المرن في حالة استخدامك لنظام التشغيل من قرص مرن أو على القرص الصلب في حالة استخدامك لنظام التشغيل من قرص صلب.

الموضوع الثاني: تهيئة قاعدة البيانات

Configuring dBASE III PLUS

المقصود بتهيئة قاعدة البيانات هو التوافق بين قاعدة البيانات واستخداماتك الخاصة. ولكي يتم ذلك يجب أن تنشئ ملفًا خاصًا اسمه CONFIG.DB هذا الملف يشتمل على مجموعة من الأوامر تنفذ تلقائيًا في كل مرة يبدأ فيها تشغيل قاعدة البيانات.

فمثلاً تستطيع تخصيص أوامر أو قيم معينة لمجموعة مفاتيح الوظائف أو تخصيص الألوان التي تروق لك للشاشة التي تتعامل معها أو تخصيص منسق للكلمات غير الذي تستخدمه قاعدة البيانات لتكتب عليه برامجك في حالة استخدامك الأمر:

MODIFY COMMAND

(١) BUFFERS:

مساحة داخل ذاكرة الحاسب تستخدم مؤقتًا لمعالجة البيانات وعادة تحجز في شكل Blocks. ويتحدد إجمالي عدد المساحات التي يمكن حجزها داخل ذاكرة الحاسب تبعًا لحجم الذاكرة المتاحة والحد الأقصى المسموح هو ٩٩.

وكذلك تخصيص حد أقصى لعدد المتغيرات التي توضع في ذاكرة الحاسب . . . إلخ .
وعادة يوضع هذا الملف على القرص الأول من أقراص نظام قاعدة البيانات
بأمرين ابتدائيين هما :

STATUS = ON

COMMAND = ASSIST

ويعني الأمر الأول إظهار سطر الحالة أسفل الشاشة دائماً . أما الأمر الثاني فمعناه توجيه قاعدة البيانات لتبدأ العمل من خلال شاشة المساعدة (Assist Menu) وطبعاً بإمكانك إجراء التعديلات التي تراها لتناسب استخدامك الخاص . فمثلاً في حالة إلغاء أو حذف الأمر الثاني فإن قاعدة البيانات ستبدأ العمل من نقطة توجيه الأوامر (Dot Prompt) بدلاً من شاشة المساعدة وبالتالي إذا كنت ترغب في تخصيص وحدة قرص صلب لتضع عليها برامجك وملفاتك فيمكنك تخصيص الوحدة التي ستخصص تلقائياً معك في ملف CONFIG. DB بالأمر DEFAULT = C وتستطيع أن تضع في هذا الملف العديد من الاختيارات ويجب أن تعرف أيها يناسب استخدامك الخاص .

وإليك مثلاً كاملاً لمحتويات هذا الملف يمكن الاسترشاد به عند تصميمك

للملف :

COLOR	= W + /B + , GR + /R
COMMAND	= CLEAR
DEFAULT	= C
ECHO	= OFF
PATH	= C: \DBASE3\DBAPP
PROMPT	= dBcmd?
TALK	= OFF
TEDIT	= \SPFPC\SPFPC
WP	= WS

وإليك شرح المصطلحات والمعطيات الموجودة في هذا المثال :

١ - الأمر COLOR = W + / B + , GR + / R

معناه أن تبدأ قاعدة البيانات بأمامية بيضاء (لون الكتابة) وخلفية زرقاء (لون الشاشة). وحروف صفراء مع خلفية حمراء في حالة استخدامك للشاشة العكسية .
(Reverse Vedio)

٢ - الأمر COMMAND = CLEAR

يوجه قاعدة البيانات لتبدأ من نقطة الموجه بشاشة خالية .

٣ - الأمر DEFAULT = C

معناه تخصيص مشغل القرص C لتسجيل البيانات عليه أو البحث فيه عن الملف المطلوب .

٤ - الأمر ECHO = OFF

يضع حالة ترديد الأوامر في وضع لا .

٥ - الأمر PATH = C:\DBASE3\DBAPP

معناه توجيه قاعدة البيانات للبحث في الدليل الفرعي DBAPP الموجود تحت الدليل DBASE3 وذلك من خلال القرص C عن الملفات التي قد لا تجدها في الدليل المخصص معها .

٦ - الأمر PROMPT = dBcmd?

يعني إجبار قاعدة البيانات أن تظهر رسالة التوجيه dBcmd? بدلاً من النقطة التي تظهر دائماً (.) .

٧ - الأمر TALK = OFF

يضع حالة إظهار نتائج الأوامر مباشرة على الشاشة في وضع لا .

٨ - الأمر TEDIT\SPFPC\SPFPC

يوجه قاعدة البيانات إلى كاتب النصوص المسمى SPFPC والموجود تحت الدليل SPFPC في حالة إدخال الأمر MODIFY COMMAND وقد يكون ذلك مفيداً

في حالة البرامج الكبيرة التي لا يتحملها منسق النصوص الموجود مع قاعدة البيانات أو في حالة الحاجة إلى كتابة نصوص عربية داخل البرنامج لأن منسق الكلمات الموجود مع قاعدة البيانات لا يصلح لذلك.

$$٩ - \text{ الأمر } WP = WS$$

عند استخدامك لحقل ملاحظات (Memo Field) داخل ملف قاعدة البيانات، فإن الكتابة أو التعديل في هذا الحقل تتم بواسطة منسق للكلمات مخصوص موجود مع قاعدة البيانات.

إذا أردت استبداله بمنسق كلمات آخر وليكن WordStar، استخدم هذا الأمر.

ملاحظة هامة:

في حالة إضافة أو تعديل أي أوامر داخل ملف CONFIG. DB فإن قاعدة البيانات لن تتأثر بالتعديلات الجديدة إلا في المرة القادمة التي يتم فيها تشغيل قاعدة البيانات.

التمتع ببرنامج آخر

بالرغم من الإمكانيات الهائلة والشهرة العظيمة التي تتمتع بها «دي بيس ثري بلاس» إلا أنها لا تستطيع أداء كل شيء بمفردها ويلزمك دائماً التعامل مع حزم برامجية أخرى للحصول على تسهيلات لا توجد بها وأكبر مثال على ذلك إعداد الرسوم البيانية فكلما هو معروف أن قاعدة البيانات «دي بيس» تعاني قصوراً شديداً في هذه الناحية.

إلا أنها تمتاز بالتوافق مع كثير من الحزم البرامجية الأخرى الموجودة في السوق. مثل برامج قوائم البيانات وبرامج الجداول الإلكترونية (Spread sheets) وبرامج معالجة النصوص (Word processor) وبرامج الرسوم البيانية (Graphics). فتستطيع قاعدة البيانات قراءة بيانات مكتوبة ببرامج معالجة الجداول الإلكترونية المشهورة مثل «لوتس ١ - ٢ - ٣» أو «سيمفوني». كما تستطيع قراءة ملفات DIF أو SYLK. وهذين النوعين من الملفات تخزن بياناتهما في شكل جداول تشبه جداول «لوتس ١ - ٢ - ٣».

وأيضاً يمكن تحويل ملفات قاعدة البيانات إلى الشفرة الأمريكية القياسية (ASCII) أو قراءة ملفات مكتوبة بهذه الشفرة وهي الشفرة التي تقرأها معظم البرامج .

استخدام محرر سطور لكتابة البرامج

بالرغم من أن «دي بيس ثري بلاس» تشتمل على محرر سطور يستخدم لكتابة البرامج وتعديلها إلا أننا ننصح دائماً بالابتعاد عنه واستخدام محرر للسطور غيره من البرامج المعروفة في السوق والتي تروق لك . وذلك لعدة اعتبارات هي :

— أن محرر السطور الذي نستخدمه «دي بيس» لا يستطيع التعامل مع برنامج يزيد طوله عن ٤ كيلوبايت .

— يعتره قصور شديد مما يصعب كتابة البرامج باستخدامه فمثلاً لا يمكن حذف أو نقل أو نسخ أكثر من سطر واحد في المرة الواحدة .

— إذا كنت تنوي استخدام أحد برامج التعريب المتاحة مثل نافذة أو مساعد العربي فهو غير صالح لتعديل البرامج بعد كتابتها أول مرة .

ولتفادي هذه العيوب ينصح بكتابة البرامج بمحرر آخر للسطور وتنفيذها فقط تحت «دي بيس» . ويجوز أن تربط أي محرر للسطور مع «دي بيس» بحيث يتم التعامل معه مباشرة بمجرد استدعاء أمر `MODIFY COMMAND <program file>` وذلك بتعديل ملف `CONFIG.DB` كما مرّ بنا فمثلاً إذا كنت تريد كتابة برامجك باستخدام معالج الكلمات `Wordstar 2000` ضع في ملف `CONFIG.DB` هذا الأمر :

`TEDIT = WS`

وأما إذا كنت تريد كتابة البرامج باستخدام المعالج `SPFPC` استخدم الأمر التالي :

`TEDIT = SPFPC`

ويجب أن تتأكد أن معالج الكلمات موجود على نفس الدليل (Directory) الذي توجد عليه قاعدة البيانات . وإلا فيجب أن تحدد اسم الدليل ثم اسم معالج الكلمات الذي ستستخدمه . ففترض أن برنامج «وورد ستار» موجود تحت دليل آخر اسمه `WORD` فيجب أن يتعدل الأمر ليصير هكذا :

`TEDIT = \WORD\WS`

وننصحك باستخدام محرر للسطور (Editor) وليس معالج للكلمات (Word Processor) لأن محرر السطور عادة أسهل في تعلمه ويشغل مساحة أقل على القرص الممغنط وأنا شخصياً أستخدم المعالج SPFPD في كتابة برامجي . ويمكنك استخدام غيره من البرامج السهلة والقوية مثل :

BRIEF _ VEDIT PLUS _ Norton Editor

التعامل مع ذاكرة الحاسب

تستخدم ذاكرة الحاسب كمخزن مؤقت توضع به البيانات المطلوب معالجتها داخل البرامج . وعادة يخصص لكل مخزن اسم مختلف عن غيره داخل الذاكرة وتوضع به بيانات حرفية أو رقمية أو تاريخية أو منطقية . وتستخدم قاعدة البيانات اصطلاح Memory Variable لكل مخزن أو موقع داخل ذاكرة الحاسب توضع به بيانات . ونحن نقول عنها تجاوزاً «حقول ذاكرة» جريئاً على تسمية حقول الملف لأنها تشبهها في أوجه كثيرة غير أن حقول الملف توضع داخل ملف قاعدة البيانات على القرص الممغنط بينما توضع حقول الذاكرة داخل ذاكرة الحاسب ويتم إعلان حقول الذاكرة بطريقتين :

١ - عامة PUBLIC

٢ - خاصة PRIVATE

١ - حقول الذاكرة العامة Public Memory Variables

وحقول الذاكرة العامة تكون متاحة لجميع البرامج في جميع المستويات حتى يتم حذفها أو إلغاؤها من الذاكرة بأمر RELEASE أو CLEAR ALL وتفهم قاعدة البيانات أن حقول / حقول الذاكرة PUBLIC بإصدار الأمر الآتي :

PUBLIC <memvar list>

في بداية أي برنامج من برامج النظام وفي أي مستوى منه .

٢ - حقول الذاكرة الخاصة Private Memory Variables

يتم إنشائها بأمر:

STORE<exp> To <memvar>

أو أمر:

<memvars> = <exp>

وتكون متاحة داخل البرنامج الذي تنشأ بداخله ولجميع البرامج التي يستدعيها هذا البرنامج في المستويات الأدنى وتحذف من الذاكرة بمجرد إنهاء البرنامج الذي أنشأها.

ويفضل أن تعلن حقول الذاكرة التي تستخدم داخل برنامج واحد فقط على أنها PRIVATE حتى لا تحدث مشكلة إذا استخدمت نفس الاسم داخل برنامج وحدث أن استدعى هذا البرنامج برنامجاً يستخدم نفس اسم المتغير - لأن الأخير سيحل محل الموجود في البرنامج الرئيسي . وتفهم قاعدة البيانات أن حقل / حقول الذاكرة خاصة PRIVATE بإصدار الأمر الآتي في بداية البرنامج .

PRIVATE <memvar list>

قبول متغيرات من خارج البرنامج

يمكن قبول حقول ذاكرة أو متغيرات من خارج البرنامج بالأمر الآتي :

DO <Program> WITH <Parm1>, <Parm2>...

ويشترط أن يشتمل البرنامج المطلوب للتنفيذ على الأمر الآتي في أول سطر:

PARAMETERS <Parm1>, <Parm2>

ويشترط كذلك أن يكون عدد المتغيرات متساوياً في الأمرين لأن قيمة <Parm1> في أمر

DO ستخصص للمتغير <Parm1> في أمر PARAMETERS وقيمة <Parm2> في أمر

DO ستخصص للمتغير <Parm2> في أمر PARAMETERS . . . وهكذا.

وتحقق هذه الطريقة ميزة كبرى في البرامج التي تنفذ أكثر من مرة وتستخدم قيمها مختلفة أثناء التنفيذ من مرة لأخرى.

حفظ محتويات الذاكرة على ملف

يمكن حفظ حقول الذاكرة الموجودة بالذاكرة على ملف خارجي على القرص الممغنط بالأمر الآتي :

SAVE TO <memory file>>

ونلجأ إلى ذلك إذا كنا سنستخدم نفس حقول الذاكرة بنفس القيم الموجودة بها فيما بعد لأن محتويات الذاكرة تحذف بمجرد إغلاق قاعدة البيانات بأمر Quit . ونحفظ محتويات الذاكرة على ملف يخصص له الاسم الممتد MEM .

ويمكن اختيار بعض حقول الذاكرة لحفظها على الملف إذا استخدم الأمر بالصيغة التالية :

SAVE ALL LIKE <skeleton> TO <memory file>

أو

SAVE ALL EXCEPT <skeleton> TO <memory file>

استرجاع محتويات الذاكرة

يستخدم الأمر الآتي لاسترجاع محتويات الذاكرة التي تم حفظها على ملف ذاكرة (MEM).

RESTORE FORM <memory file> [ADDITIVE]

وإذا استخدم الأمر بدون الاختيار ADDITIVE فستحل حقول الذاكرة محل الموجود بالذاكرة وستحوي تلك الموجودة من قبل . أما إضافة الاختيار ADDITIVE فإنها تضيف حقول الذاكرة المسترجعة من الملف إلى تلك الموجودة بداخل الذاكرة .

ومن الأمثلة التي نحتاج فيها لحفظ واسترجاع حقول الذاكرة إلى أو من ملف خارجي حالة إصدار فواتير بيع بأرقام متسلسلة فيمكن حفظ رقم آخر فاتورة أصدرت على ملف خارجي في نهاية اليوم ويتم استرجاع هذا الرقم في بداية اليوم التالي ليتولى برنامج إصدار الفواتير إضافة واحد لرقم آخر فاتورة أصدرت .

الفصل الثالث

إعداد النظام

يوضح هذا الفصل فكرة النظام الذي سنشرحه في هذا الكتاب. والنظام في الحقيقة عبارة عن ستة أنظمة يمكن فصل أى واحد منها وتشغيله منفردا أو بيعه في الأسواق سواء بحالته أو بعد توفيقه مع رغبات العميل. والأنظمة الستة يمكن تشغيلها من داخل قائمة واحدة. وفي هذا الفصل سنشرح هذه القائمة والبرامج التي تشمل عليها.

قبل أن نبدأ في مراحل إعداد النظام المقترح أقترح أن ننشئ دليلًا فرعيًا خاصًا لتضع عليه نظام إدارة قاعدة البيانات ويفضل أن يوضع تحت الدليل الخاص بقاعدة البيانات. والغرض من إنشاء هذا الدليل الفرعي أن تحتفظ بملفاتك ويرايجك مستقلة عن ملفات البرامج الخاصة بقاعدة البيانات والموجودة تحت دليل قاعدة البيانات وبذلك يمكنك بسهولة نقل ملفات هذا النظام إلى دليل فرعي آخر أو إلى قرص مرن.

أنشئ دليلًا فرعيًا باسم DBAPP تحت دليل قاعدة البيانات، بفرض أن اسم دليل قاعدة البيانات هو DBASE3 بالأمير الآتي من تحت نظام التشغيل:

C:\>MD\DBASE3\DBAPP

ثم انتقل إلى هذا الدليل قبل بداية العمل لكي تضمن أن الملفات التي ستنشئ ستوضع تحت هذا الدليل. ولكي تنتقل إلى هذا الدليل أدخل الأمر التالي تحت تحت نظام التشغيل.

C:\>CD\DBASE3\DBAPP

ستظهر رسالة المحث هكذا:

C:\DBASE3\DBAPP>

ملاحظة: إذا لم يظهر اسم الدليل الحالي ضمن رسالة المحث أدخل الأمر الآتي من تحت نظام التشغيل

C.\>PROMPT \$P \$G

لتشغيل قاعدة البيانات من هذا الدليل اكتب DBASE ثم اضغط مفتاح الإدخال هكذا:

C:\DBASE3\DBAPP>DBASE

ويشترط لذلك أن يشتمل أمر PATH على اسم الدليل الذي توجد تحته قاعدة البيانات.

بهذه الطريقة تضمن أن الملفات التي ستنشئ ستوضع تحت الدليل الفرعي DBAPP الموجود تحت الدليل DBASE3 فإذا لم ترغب في استخدام أمر PATH أو لم ترد تشغيل قاعدة البيانات من الدليل الفرعي DBAPP فيمكنك تشغيل قاعدة البيانات

بالطريقة التي تعرفها (بعد أن تنشئ الدليل الفرعي طبعاً) ثم تدخل الأمر الآتي من نقطة توجيه الأوامر

.RUN CD DBAPP

ومهمة هذا الأمر هي الانتقال إلى الدليل الفرعي DBAPP تحت دليل قاعدة البيانات وبالتالي وضع الملفات التي ستنشأ تحت هذا الدليل والبحث تحته عن الملفات المطلوبة .

فكرة النظام

شركة الحاسب العربي تعمل بمجال تجارة الحاسبات وهي تشتري البضاعة باعتمادات مستندية من الخارج وقد تصل البضاعة مرة واحدة أو على دفعات وعندما تصل البضاعة يتم تسجيلها بالدفاتر والمخازن . ويظل الاعتماد مفتوحاً حتى تصل آخر دفعة من البضاعة ثم يغلق . ويحتاج مدير الشركة من حين لآخر لمعرفة تقارير عن الاعتمادات أو اعتمادات مختارة أو عن فواتير الشراء . ويطلب من حين لآخر تقارير عن حركة المخزون والأصناف التي قاربت نقطة إعادة الطلب . وحركة المشتريات يطلب المدير أيضاً من حين لآخر تقارير عن المبيعات وحسابات العملاء ويريد المدير استبدال نظام الاعتمادات المستندية والمشتريات والمبيعات والمخازن وحسابات العملاء من النظام اليدوي لنظام آلي يعمل بالحاسب .

نفرض أننا تقابلنا مع مدير الشركة وعرفنا طلباته واطلعنا على مستنداته ودفاتره . مطلوب تطوير النظام أو النظم التي تخدم كل الأغراض المطلوبة .

سنتناول في هذا الفصل والفصول القادمة شرح كيفية تطوير هذا النظام . والنظام في الحقيقة يشتمل على عدة أنظمة . يمكن أن يكون كل منها منفصلاً عن باقي الأنظمة كما يمكن تجميع هذه الأنظمة وتشغيلها من قائمة رئيسية للنظام .

سنحاول في هذا الفصل والفصول التالية توضيح خطوات تصميم النظام والملفات والبرامج وشرح كل برنامج على حدة وفي النهاية سنكون غطينا - بإذن الله -

جميع إمكانيات واستخدامات قاعدة البيانات dBASE III Plus وأوضحنا جميع التسهيلات الموجودة بها وكيفية الاستفادة منها.

ويمكن الاستفادة من التطبيقات التي يشتمل عليها هذا الفصل والفصول التالية في جوانب كثيرة منها:

أولاً: لتوضيح تسهيلات وإمكانيات قاعدة البيانات وكيفية الاستفادة منها من خلال كمية كبيرة من البرامج مرتبطة مع بعضها داخل نظام متكامل.

ثانياً: تطبيقات يستفاد منها بحالتها الراهنة لمراقبة المخزون وإصدار فواتير البيع والشراء والحصول على تقارير مفصلة عن حركة المخزون والمبيعات والمشتريات والاعتمادات. وأيضاً لمراقبة حسابات العملاء والحصول على تقارير مفصلة عن عميل أو كل العملاء أو عينة مختارة من العملاء أو عن الحسابات وحركة المدين والدائن منها.

ثالثاً: تطبيقات يستفاد بها عموماً في التطبيقات الأخرى بطريقة إعداد القوائم الرئيسية والفرعية والفواتير والتقارير وصيانة الملفات لاختلاف من تطبيق لآخر.

تطوير النظام من النظم

النظام الذي ننوي تطويره يخدم الأغراض الآتية.

- * صيانة ملفات الاعتمادات المستندية بالإضافة أو التعديل أو الحذف.
- * استخراج التقارير المطلوبة عن اعتماد برقمه أو اعتمادات خلال مدة معينة.
- * صيانة ملف المخزون بالإضافة أصناف جديدة كلما طرأت حاجة أو حذف أصناف توقف التعامل بها أو تعديل بيانات صنف موجود.
- * استخراج تقارير عن الأصناف الموجودة وأرصدها والأصناف التي وصلت إلى نقطة إعادة الطلب.
- * تسجيل فواتير الشراء وتعديل أرصدة الأصناف المشتراة بالإضافة الكميات الجديدة إلى الرصيد الحالي لكل صنف واستخراج التقارير اللازمة عن فاتورة أو فواتير الشراء خلال مدة معينة.

* تسجيل فواتير المبيعات وتعديل أرصدة الأصناف المباعة بخضم الكميات المباعة من الرصيد الحالي لكل صنف واستخراج التقارير اللازمة عن فاتورة أو فواتير البيع خلال مدة معينة .

* مراقبة حسابات العملاء وتشمل إضافة عميل جديد لدى الشركة أو تعديل بيانات عميل موجود مثل تغيير عنوانه أو رقم هاتفه أو حذف عميل من ملف العملاء نظراً لانتهاؤه التعامل معه . والملف يشمل البيانات الأساسية عن كل عميل مثل رقم حسابه واسمه ثلاثياً واسم الشركة وعنوانها والمدينة التي توجد بها ورقم الهاتف والرصيد وتاريخ آخر كشف حساب ورصيد آخر كشف حساب .

* الحصول على تقارير عن العملاء أو عن عميل أو عملاء مختارين طبقاً لمعايير مختلفة . وكذلك استخراج ملصقات لتطبع على مراسلاتهم .

* الحصول على تقارير عن الحسابات وحركة كل منها وطباعة كشوف الحساب الشهرية .

* بالإضافة إلى الوظائف السابقة لابد من إيجاد وسيلة لصيانة ملفات النظام كله تتمثل في :

- عمل نسخ احتياطية للملفات على أقراص لينة تحسباً لأخطار التلف أو الحريق . . . أو غيرها .
- إمكانية استرجاع النسخ الاحتياطية كلما لزم الأمر .
- فهرسة أو إعادة فهرسة الملفات في أي وقت .
- تنظيف الملفات من السجلات الغير مطلوبة .
- إعادة ترصيد الكميات الموجودة بالمخازن بعد عمليات الشراء أو البيع التي تتم .

ولأن هذه الوظائف كثيرة وتشتمل في الحقيقة على أكثر من نظام فستلجأ إلى عدة

أنظمة تشمل ما يلي :

- | | |
|-------------------------------|------------------------|
| ١ - طلبات الشراء (الاعتمادات) | 1 . Order of purchases |
| ٢ - المخزون | 2 . Inventory |
| ٣ - المشتريات | 3 . Purchases |
| ٤ - المبيعات | 5 . Sales |

5 . Customer Accounts

٥ - مراقبة حسابات العملاء

6 . Files Maintenance

٦ - صيانة الملفات

وسنخصص لكل نظام منها فصلاً مستقلاً من الفصول التالية . وسنوضح فيما يلي القائمة الرئيسية للنظام التي تسمح بالتعامل مع الوظائف الست السابقة والبرنامج اللازم لإظهارها .

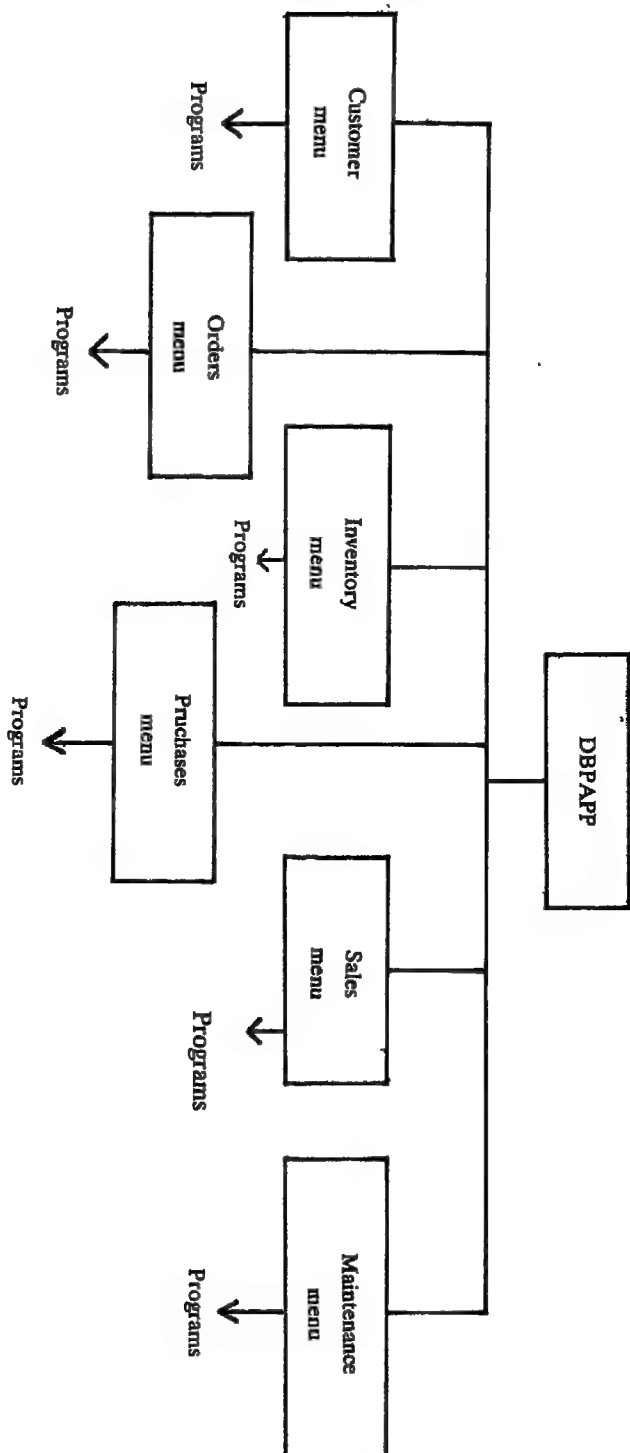
القائمة الرئيسية للنظام

تستخدم الطريقة المتبعة في تطوير هذه التطبيقات مفهوم البرمجة التركيبية - Structured Programming وتعتمد على تطوير عدد كبير من البرامج والبرامج الصغيرة تسمى Modules . ويقوم كل برنامج بوظيفة محددة . وتتعاون مجموعة من البرامج في أداء العمل المطلوب . والغاية من ذلك تسهيل إعداد هذه التطبيقات بالإضافة إلى تسهيل تعديلها أو صيانتها في المستقبل إذا لزم الأمر . ومن ناحية أخرى كلما كان البرنامج صغيراً كلما سهل تعقب واكتشاف أخطائه في مرحلة التجارب .

وسنبداً في شرح النظام من أعلى إلى أسفل ويسمى هذا المفهوم في تحليل الأنظمة top-down design ولذلك سنبداً برنامج القائمة الرئيسية للنظام .

يقوم البرنامج الرئيسي في النظام بربط الأنظمة الأخرى مع بعضها بحيث يمكن الوصول إلى أي منها من خلال قائمة واحدة ويوضح شكل ١ - ٣ الخريطة الرئيسية للنظام والقوائم التابعة لها بصفة إجمالية لأن كل قائمة منها ستناقش في فصل مستقل .

ويسمى هذا الشكل بالنظام الهرمي Hierarchical system ويمتاز بالبساطة والوضوح وسهولة إضافة أو حذف اختيار إليه ولذلك فإن العميل لن يبذل جهداً كبيراً لفهم النظام وتعلمه إلا أن هذا النظام لايسمح بالانتقال مباشرة من برنامج تابع لإحدى القوائم إلى آخر تابع لقائمة أخرى . فمثلاً لا يمكنك الانتقال مباشرة من برنامج داخل قائمة المشتريات إلى آخر داخل قائمة المبيعات . وإنما الذي يحدث أنك



شكل ١ - ٣ الخريطة الرئيسية للنظام والوقائيم التابعة لها

ترجع إلى القائمة الرئيسية بعد الانتهاء من برنامج داخل قائمة المشتريات لتختار قائمة المبيعات من القائمة الرئيسية ومن ثم البرنامج المطلوب منها .

ويوضح شكل ٢ - ٣ القائمة الرئيسية للنظام كما ستظهر للمستخدم للاختيارات التي تشتمل عليها وهي ترجمة للخريطة الموجودة في شكل ١ - ٣ .

Friday 1.March 1991
Time: 11:59:49

Arabian Computer Center
C.R. 39387

Main menu

- (1) Orders of purchases
- (2) Inventory.....
- (3) Purchases.....
- (4) Sales
- (5) Customer Account...
- (6) Files maintenances..
- (X) Exit to Dot-Prompt.
- (Q) Quit to DOS.....

Select number or colored letter ==> █

شكل ٢ - ٣ القائمة الرئيسية للنظام

البرنامج الرئيسي

قبل أن نتكلم عن البرنامج اللازم لإظهار هذه القائمة وتنفيذ اختياراتها نود أن نوضح أننا سنعمد أثناء شرح البرامج التي يشتمل عليها هذا الكتاب على أن القارئ ذو خبرة سابقة بقاعدة البيانات dBASE III PLUS . وأوامرها وأنه انتهى من كتابنا السابق المرجع الأساسي لقاعدة البيانات dBASE III PLUS الذي يشرح أساسيات العمل بها وكيفية كتابة برامجها . ولذلك فسنشير إلى فكرة البرنامج أو جزء البرنامج بصفة عامة ثم نتناول بالشرح الأوامر والمفاهيم بطريقة مختصرة اعتماداً على خبرة القارئ السابقة بالأوامر والمفاهيم البديهية . كما أن الفكرة أو الأوامر الجديدة التي سنشرحها في

أحد البرامج سنغفلها إذا وردت في برنامج لاحق اعتماداً على أننا شرحناها في مرة سابقة إلا إذا دعت ضرورة إلى ذلك .

يشتمل شكل ٣ - ٣ على برنامج DBAPP.PRG وهو البرنامج اللازم لإظهار القائمة الموجودة في شكل ٣ - ٢ ، وتلاحظ في شكل ٣ - ٣ ما يلي :

```

1 * ----- *
2 * Program.....: DBAPP.PRG *
3 * Programmer...: Magdi M. Abu Al-Ata *
4 * Date.....: Dec. 25,1990 *
5 * Purpose.....: To display main menu and do selected option *
6 * Copyright...: Magdi Mohammad Abu Al-Ata *
7 * ----- *
8 SET PROCEDURE TO dbproc
9 PRIVATE scrttitle,scrfootr    && Don't use their values in other modules
10 DO init
11 scrttitle = "Main menu"
12 scrfootr = "Select number or colored letter ==>"
13 ok = .T.
14 DO WHILE ok
15     choice = " "
16     DO border WITH scrttitle,scrfootr    && Do screen border
17     * Draw menu
18     @10, 5 SAY "(1) Orders of purchases "
19     DO intense WITH 10,10,"O"
20     @11, 5 SAY "(2) Inventory....."
21     DO intense WITH 11,10,"I"
22     @12, 5 SAY "(3) Purchases..... .."
23     DO intense WITH 12,10,"P"
24     @13, 5 SAY "(4) Sales ....."
25     DO intense WITH 13,10,"S"
26     @14, 5 SAY "(5) Customer Account..."
27     DO intense WITH 14,10,"C"
28     @15, 5 SAY "(6) Files maintenance.."
29     DO intense WITH 15,10,"F"
30     @16, 5 SAY "(X) Exit to Dot-Prompt."
31     DO intense WITH 16,11,"X"
32     @17, 5 SAY "(Q) Quit to DOS....."
33     DO intense WITH 17,10,"Q"
34     @22,43 GET choice
35     READ
36     * Do selected option
37     DO CASE
38     CASE choice = "1" .OR. UPPER(choice) = "O"
39     | DO ordmenu    && Purchase orders options
40     CASE choice = "2" .OR. UPPER(choice) = "I"
41     | DO invmenu    && Inventory options

```

شكل ٣ - ٣ برنامج DBAPP.PRG

```

42 | CASE choice = "3" .OR. UPPER(choice) = "p"
43 |     DO purmenu                && Recording received items
44 | CASE choice = "4" .OR. UPPER(choice) = "s"
45 |     DO salmenu                && Process sales invoices
46 | CASE choice = "5" .OR. UPPER(choice) = "c"
47 |     DO cusmenu                && Process customer data
48 | CASE choice = "6" .OR. UPPER(choice) = "f"
49 |     DO mntmenu                && Maintenance options
50 | CASE UPPER(choice) = "x"
51 |     * Restore system environments before exit
52 |     DO reset
53 | <-----RETURN                && Go to Dot-Prompt
54 | CASE UPPER(choice) = "q"
55 |     CLEAR
56 | <=====QUIT                && Quit to DOS
57 | OTHERWISE
58 | <-----LOOP
59 | ENDCASE
60 ENDDO
61 *****
62 * End of file DBAPP.PRG      *
63 *****

```

تابع شكل ٣ - ٣ برنامج DBAPP.PRG

□ استخدمنا علامة * في بداية السطر أو علامات & & بعد الأوامر بقدر المستطاع لتوضيح المقصود من هذه الأوامر ولتكون بديلاً عن الشرح الوارد في الكتاب للأوامر السهلة والمباشرة.

□ وضعنا أرقاماً متسلسلة أمام أوامر البرنامج لأننا سنستخدم هذه الأرقام للإشارة إلى الأمر الذي يتبعها أثناء الشرح بدلاً من إعادة كتابة الأمر مرة ثانية.

□ رسمنا خطأ رأسياً متصلاً أمام كل مجموعة أوامر ذات الصلة داخل البرنامج. فمثلاً، رسمنا هذا الخط أمام الأوامر التي تقع داخل دالة واحدة، وكذلك للإشارة إلى الأوامر التي ستنفذ إذا وقعت حالة معينة صحيحة. . . وهكذا.

□ الأوامر التي تسبب تفرعاً داخل البرنامج أو تغيير في ترتيب تنفيذ تعليمات البرنامج عما هي عليه داخل البرنامج رسمنا أمامها سهمًا ليوضح اتجاه التفرع نتيجة تنفيذ أحدها فمثلاً أمر RETURN (أو QUIT) داخل الدالة يبدو هكذا.

RETURN

وهذا يعني أن هذا الأمر سيسبب الخروج من الدوارة أما السهم الموجود أمام أمر LOOP داخل الدوارة فيبدو هكذا

↑ LOOP

وهذا يعني أن الأمر سيسبب الرجوع إلى بداية الدوارة في اتجاه بداية البرنامج .

والغرض من ذلك زيادة وضوح البرنامج للقارئ ومتابعة منطقة بمجرد النظر وستتبع هذه الطريقة عند شرح باقي برامج الكتاب .

والآن نعود لشرح محتويات هذا البرنامج .

- يبدأ البرنامج في السطر رقم ٨ بفتح ملف الإجراءات DBPOROC أي نقله من القرص المغنط إلى الذاكرة (سنشرح استخدام ملف الإجراءات بعد شرح هذا البرنامج).

- يخصص الأمر رقم ٩ في بداية البرنامج حقول الذاكرة المذكورة لهذا البرنامج فقط لأننا سنستخدم نفس الأسماء في برامج أخرى داخل النظام .

- يتم استدعاء إجراء موجود داخل ملف الإجراءات والإجراء اسمه init (سطر رقم ١٠) ومهمته تخصيص القيم التي تتحكم في طريقة عمل البرنامج (سنناقش هذا الإجراء بعد شرح البرنامج).

- يقع جسم البرنامج بين الدوارة DO WHILE ... ENDDO ومعناها أن الأوامر الواقعة بينها ستكرر طالما أن محتويات OK هي T .

- تلخص وظيفة هذه الدوارة في إظهار اختيارات القائمة وقبول قيمة من المستخدم لتنفيذ الاختيار المطلوب . وفي داخل هذه الدوارة يتم استدعاء الإجراء intense بعد كتابة كل اختيار لكتابة الحرف المختار بلون مخالف (سنناقش الإجراء intense بعد الانتهاء من شرح البرنامج).

- يسمح البرنامج بتنفيذ واحدة من عدة حالات بناء على اختيار المستخدم ويتسبب كل اختيار في استدعاء برنامج معين يؤدي وظيفة محددة (سطر رقم ٣٧) .

- يتم تنفيذ أحد الاختيارات إما باختيار الرقم الذي أمامه أو الحرف الملون (سطر رقم ٣٨) . وقد استخدمنا الوظيفة UPPER (choise) لإعطاء المستخدم حرية أكثر عند

- كتابة الحرف الدال على اختياره بالحرف الكبير (Upper) أو الصغير (Lower) لأن الوظيفة تحوله دائماً إلى حرف كبير.
- يستمر ظهور القائمة بعد الانتهاء من تنفيذ أحد الاختيارات حتى يقرر المستخدم الخروج بإحدى طريقتين:
- ١ - اختيار حرف X (سطر رقم ٥٠) وهو يسبب استدعاء الإجراء reset لإعادة الدوال التي خصصها الإجراء init في أول البرنامج إلى حالتها السابقة ثم الخروج من الدوارة وبالتالي إنهاء البرنامج. (سنشرح الإجراء reset بعد قليل).
 - ب - اختيار حرف () (سطر رقم ٥٤) وهو يسبب إنهاء البرنامج والعودة إلى نظام التشغيل.
- إذا اختار المستخدم اختياراً آخر غير تلك الموضحة بالقائمة ينتقل التنفيذ إلى بداية الدوارة (سطر رقم ٥٧ ، ٥٨).

استخدام برنامج الإجراءات (Procedure file)

شرحنا بالتفصيل إنشاء واستخدام ملف الإجراءات والقيود الواردة عليه في الفصل الأول. ورغم أننا نصحن أن الملف يجب أن يعد بعد الانتهاء من النظام واختباره فإننا استخدمناه هنا لأننا جربنا النظام فعلاً قبل شرحه في هذا الكتاب. ويمكنك وضع كل إجراء داخل برنامج مستقل. وفي هذه الحالة سيتم استدعاء البرنامج كما لو كان إجراء داخل ملف إجراءات وبعد الانتهاء من دراسة النظام انقل هذه البرامج الصغيرة إلى ملف الإجراءات لتستفيد من المزايا التي شرحناها في الفصل الأول.

سنضع في ملف الإجراءات DBPROC.PRG البرامج الصغيرة والتي تستدعي لأكثر من مرة داخل النظام ولأننا يمكننا أن نضع به حتى ٣٢ إجراءً. فيمكنك إضافة البرامج التي تراها إليه بعد الانتهاء من دراسة النظام. ولن نتعرض لشرح برنامج DBPROC.PRG مرة واحدة. وإنما سنشرح الإجراء الذي يلزمنا داخل البرنامج الذي

نتناوله. وفي النهاية سنكون شرحنا جميع الإجراءات الموجودة داخل برنامج .
DBPROC.PRG

يشتمل البرنامج الرئيسي DBAPP.PRG الذي بين أيدينا على أربعة إجراءات
موجودة بملف DBPROC.PRG وهي :

١ - الإجراء *init*

يشتمل شكل ٤ - ٣ على الإجراء Init وعن هذا الإجراء نوضح ما يلي :
- الأمر الموجود في سطر رقم ١٢ يعلن بداية الإجراء داخل ملف الإجراءات واسمه .

```
8 *****
9 * Procedure : init
10 * Purpose : Set working environments
11 *****

12 PROCEDURE init
13 PUBLIC msg,mcolor
14 SET SCOREBOARD OFF                && Turn off dBASE message in row 0
15 SET TALK OFF                      && Turn off talking
16 SET STATUS OFF                   && Turn off status bar
17 SET CONFIRM ON                   && Press Enter to accept entry
18 SET BELL OFF                     && Turn bell sound off
19 SET EXACT ON                     && Turn the bell off
20 SET DATE BRITISH                 && Display date in format: dd/mm/yy
21 SET DELETED ON                   && Ignire records marked for deletion
22 CLOSE ALL                        && Close all files
23 msg = " Press " + chr(17)+chr(196)+chr(217) + "key to try again."
24 IF ISCOLOR()
25 |   mcolor = .T.
26 ENDIF
27 IF mcolor
28 |   SET COLOR TO W+/N,W+/G+
29 ENDIF
30 RETURN                            && End of procedure
31
```

شكل ٤ - ٣ الإجراء *init*

- سطر رقم ١٣ معناه إعلان تخصيص كل من msg و mcolor لكل برامج النظام .
- الأوامر الموجودة بالسطور من ١٤ إلى ٢١ لتخصيص القيم الابتدائية المطلوبة للنظام
والتي تتحكم في طريقة عمل البرنامج وتفصيلها كما يلي :

- سطر رقم ١٤ لإلغاء الرسالة التي تظهرها قاعدة البيانات في السطر الأول من الشاشة (سطر رقم صفر) نتيجة ضغط مفتاح Ins أو Caps مثلاً حتى لا تغطي هذه الرسالة البيانات الموجودة في هذا السطر.
- سطر رقم ١٥ لإلغاء النتائج المباشرة لبعض الأوامر.
- سطر رقم ١٦ لإلغاء سطر الحالة الذي يظهر في أسفل الشاشة.
- سطر رقم ١٧ لكي يجبر المستفيد على ضغط مفتاح الإدخال إذا وصل إلى نهاية المساحة المخصصة لحقل الإدخال. وهذا نوع من التأكيد تبدو الحاجة إليه أكثر في حالة الحقول التي تتطلب حرفاً واحداً مثل إدخال الحرف أو الرقم الدال على اختيار معين. فإذا أخطأ المستفيد واختار حرفاً أو رقماً غير المقصود فإن اختياراً غير المقصود سينفذ. أما في حالتنا هذه فإن الاختيار لن ينفذ إلا إذا ضغط المستفيد مفتاح الإدخال بعد كتابة الحرف أو الرقم.
- سطر رقم ١٨ لإلغاء صوت الجرس الذي يسمع إذا وصل المستفيد لنهاية المساحة المخصصة لحقل الإدخال أو في حالة حدوث خطأ.
- سطر رقم ١٩ يطلب من قاعدة البيانات ضرورة مطابقة عبارة البحث مع تلك الموجودة بالملف مطابقة تامة لكي تقرر أنها موجودة.
- سطر رقم ٢٠ لتغيير شكل إظهار التاريخ هكذا: سنة / شهر / يوم.
- سطر رقم ٢١ لتجاهل قاعدة البيانات السجلات المعلمة لغرض الحذف عندما تبحث في الملف.
- سطر رقم ٢٤ معناه إذا كانت الشاشة المتصلة بالحاسب ملونة ضع القيمة المنطقية T. في حقل الذاكرة mcolor (سطر رقم ٢٥).
- سطر رقم ٢٧ معناه إذا كانت الشاشة ملونة (لأن mcolor ستشتمل على T. فقط إذا كانت الشاشة ملونة) اجعل لون الكتابة أبيض والخلفية سوداء ولون الكتابة في الشاشة العكسية أبيض والكتابة خضراء.
- سطر رقم ٣٠ في نهاية الإجراء ضروري لفهم قاعدة البيانات أن الإجراء انتهى وتعيد التنفيذ إلى البرنامج الذي استدعاه.

٢ - الإجراء reset

يشتمل شكل ٥ - ٣ على الإجراء reset وهو يعيد الدوال التي سبق تخصيصها بالإجراء init إلى حالتها السابقة، ويحذف كلا من محتويات الذاكرة والشاشة قبل إنهاء البرنامج.

```
32 *****
33 * Procedure : reset
34 * Purpose : Reset working environments and exit
35 *****

36 PROCEDURE reset
37 SET SCOREBOARD ON
38 SET TALK ON
39 SET STATUS ON
40 SET CONFIRM OFF
41 SET BELL ON
42 SET EXACT OFF
43 SET DATE AMERICAN
44 CLEAR ALL          && Erase memory
45 CLEAR              && Clear screen
46 RETURN             && End of procedure
47
```

شكل ٥ - ٣ الإجراء reset

٣ - الإجراء border

يشتمل شكل ٦ - ٣ على الإجراء border. وعن هذا الإجراء توضيح ما يلي:

- أمر PARAMETER في سطر رقم ٥٣ معناه قبول معطيات عددها ٢ من خارج البرنامج عند استدعائه للتنفيذ.
- أمر MDATE في سطر رقم ٥٧ يضع القيم التالية داخل حقل ذاكرة: اسم اليوم من التاريخ المسجل بالحاسب ثم فراغ ثم تاريخ اليوم ثم علامة "،" ثم اسم الشهر ثم السنة مكونة من ٤ خانوات.

ولأن العبارة حرفية فقد استخدمنا الوظيفة STR قبل العبارات الرقمية لتحويلها إلى عبارات حرفية. واستخدمنا وظائف التاريخ لإظهار التاريخ هكذا

3 Jan, 1991

```

48 *****
49 * Procedure : border
50 * Purpose   : Draws a screen frame
51 *****

52 PROCEDURE border
53 PARAMETERS scrttitle,scrfootr
54 PUBLIC systitle,title2,mdate,mtime
55 systitle = "Arabian Computer Center"
56 title2   = "C.R. 39387 "
57 mdate    = CDOY(DATE())+" "+STR(DAY(DATE()),2)+" "+CMONTH(DATE()) ;
57 +" "+STR(YEAR(DATE()),4)
58 mtime    = "Time: " + TIME()
59 CLEAR
60 DO intense WITH 0, 1, mdate
61 DO intense WITH 0,78 - LEN(mtime), mtime
62 IF mcolor
63 | SET COLOR TO G+/N
64 ELSE
65 | SET COLOR TO W+/N
66 ENDIF
67 @ 1, 1 TO 4,78 DOUBLE
68 @ 7, 1 TO 23,78 DOUBLE
69 @ 2,80/2-(LEN(systitle)/2) SAY systitle
70 @ 3,80/2-(LEN(title2)/2) SAY title2
71 @ 8,80/2-(LEN(scrttitle)/2) SAY scrttitle
72 @ 9,2 TO 09,77
73 @21,2 TO 21,77
74 @22,5 SAY scrfootr
75 SET COLOR TO W+/N
76 RETURN                && End of procedure
77

```

شكل ٦ - ٣ الإجراء border

بدلاً من 01/03/91 التي يظهرها النظام .

- الأمر الموجود في سطر رقم ٦٠ معناه إظهار التاريخ بلون مخالف لأن الإجراء intense

وظيفته تغيير اللون فقط وهو مساوٍ للأمر التالي إذا لم تكن بحاجة لتغيير لون الكتابة

(ii) 1, 1 SAY mdate

- السطر رقم ٦١ يظهر الوقت في آخر السطر رقم صفر وهو مساوٍ للأمر التالي إذا لم تكن بحاجة لتغيير لون الكتابة

(iii) 78 - LEN (mtime) SAY mtime

- السطر رقم ٦٢ يخصص اللون الأخضر للكتابة على الشاشة إذا كانت الشاشة ملونة . وهو خاص بالأوامر التي تظهر اسم الشركة واسم الشاشة المعروضة

- والمستطيلات والموجودة في السطور من ٦٧ إلى ٧٤.
- السطر رقم ٦٩ يضع محتويات systitle وسط السطر وبحسب بداية كتابتها كما يلي:
يقسم $2 \div 80$ ليحدد منتصف الشاشة وي طرح منه نصف طول العبارة. وبنفس الطريقة ينفذ الأمرين التاليين.
 - سطر رقم ٧٥ في نهاية الإجراء يعيد الألوان المستخدمة للكتابة على الشاشة.

٤ - الإجراء *intense*

يشتمل شكل ٧ - ٣ على الإجراء *intense* ووظيفته إظهار نص معين باللون الأصفر إذا كانت الشاشة ملونة ويقبل ٣ قيم عند استدعائه هي بالترتيب رقم السطر، رقم العمود والنص المطلوب.

```
78 *****
79 * Procedure : intense
80 * Purpose : To display text in red color
81 *****

82 PROCEDURE intense
83 PARAMETERS mrow, mcol, mstr
84 IF mcolor
85 | SET COLOR TO GR+/N
86 ELSE
87 | SET COLOR TO W+/N
88 ENDIF
89 @ mrow, mcol SAY mstr
90 IF mcolor
91 | SET COLOR TO W+/N
92 ELSE
93 | SET COLOR TO W+/N
94 ENDIF
95 RETURN
96
```

شكل ٧ - ٣ الإجراء *intense*

الفصل الرابع

الاعتمادات المستندية

Purchase orders

يشرح هذا الفصل نظام الاعتمادات المستندية أو طلبات الشراء والوظائف التي يتطلبها مثل إضافة اعتماد جديد لملف الاعتمادات المستندية أو إلغاء اعتماد أو الاستفسار منه ويشرح كذلك كيفية استخراج التقارير المطلوبة. ويعتبر هذا الفصل مقدمة لنظام المشتريات الذي سيرد في الفصل السادس. ويبدأ الفصل بشرح وظائف النظام ثم يشرح الملفات المطلوبة. ثم يشرح بالتفصيل كل برنامج من برامج النظام على هذه وعلاقته بغيره من البرامج الأخرى داخل النظام.

تشتري شركة الحاسب العربي أجهزة الحاسبات وقطع غيارها من الخارج وتحتاج لفتح اعتمادات مستندية أو طلبات شراء (purchase orders) لدى أحد البنوك حتى تصل البضاعة وتحتاج لحفظ بيانات الاعتمادات المستندية ومصرفاتها بحيث يمكنها بعد ذلك الاستفسار عن اعتماد معين أو اعتمادات خلال مدة معينة. والنظام الذي سنشرحه في هذا الفصل يقدم الوظائف التالية :

- ١ - إضافة اعتماد جديد إلى ملف الاعتمادات (طلبات الشراء).
- ٢ - تعديل بيانات الاعتماد أو تسجيل المصروفات التي تخصه بمجرد وصوله.
- ٣ - حذف بيانات اعتماد من الملف.
- ٤ - الاستفسار عن اعتماد برقمه.
- ٥ - تقارير عن الاعتمادات المستندية خلال مدة زمنية أو قبل أو بعد تاريخ معين.

ملاحظة : يمكن أن تضاف وظائف أخرى للصيانة الدورية التي يتطلبها النظام مثل عمل نسخ احتياطية للملفات أو استرجاعها في حالة الضرورة أو فهرستها . . . إلخ. ولأننا سنخصص فصل مستقل لصيانة الملفات ليخدم كل الملفات المستخدمة في الكتاب فسنرجى هذه البرامج لتشرح في حينها ويمكنك مراجعتها في الفصل التاسع. وسنوضح فيما يلي الخطوات المطلوبة لتصميم نظام يحقق كل هذه الوظائف.

تصميم ملف قاعدة البيانات

الخطوة الأولى في تصميم نظام لإدارة قاعدة البيانات هي تصميم ملف أو ملفات قاعدة لبيانات (.DBF) وملف أو ملفات الفهرسة (.NDX) المتصلة به والملف اللازم لنظام الاعتمادات المستندية هو ORDER.DBF. لإنشاء الملف المطلوب أدخل الأمر التالي من نقطة المحث:

. CREATE ORDER

ثم أدخل مواصفات الملف الموجودة في شكل ١ - ٤. ومنه يتضح أن ملف الاعتمادات المستندية أو طلبات الشراء يشتمل على البيانات الأساسية المطلوبة عن

				Bytes remaining: 3853			
Field Name	Type	Width	Dec	Field Name	Type	Width	Dec
1 ORD_NO	Character	5		16 CUSTON	Numeric	7	2
2 ORD_VAL	Numeric	7	2	17 LBR_TRNS	Numeric	7	2
3 ORD_DAT	Date	8		18 TOTAL	Numeric	7	2
4 ORD_KIND	Character	3					
5 ORD_DESC	Character	12					
6 SHIP_NAM	Character	12					
7 COMP_NAM	Character	12					
8 COMP_ADR	Character	12					
9 BNK_NAM	Character	12					
10 BNK_ADR	Character	12					
11 BILL_NO	Character	5					
12 DOC_NO	Character	5					
13 COMISION	Numeric	7	2				
14 POSTAGE	Numeric	7	2				
15 GOOD_INS	Numeric	7	2				

MODIFY STRUCTURE (C:) ORDER Field: 1/18

Enter the field name.

Field names begin with a letter and may contain letters, digits and underscores

شكل ١ - ٤ مواصفات ملف الاعتماد المستندية

الاعتماد بالإضافة إلى المصروفات التي يتطلبها حتى تصل البضاعة إلى الشركة وهذه البيانات هي :

رقم الاعتماد (ORD_NO) ، قيمة الاعتماد (ORD_VAL) ، تاريخ الاعتماد (ORD_DAT) ، نوع الاعتماد (ORD_KIND) ، فقد يكون شرط الاعتماد التسليم بلد البائع أو على ظهر الباخرة أو الدفع عند الاستلام - شرح الاعتماد (ORD_DESC) ، اسم شركة الشحن (SHIP_NAM) ، اسم الشركة البائعة (COMP_NAM) ، اسم البنك الذي سيتولى تمويل الاعتماد (BNK_NAM) ، عنوان البنك (BNK_ADR) ، رقم فاتورة الشراء (BILL_NO) ، رقم بوليصة الشحن (DOC_NO) ، العمولة (COMISION) - الدمغة (POSTAGE) ، التأمين (GOOD_INS) ، الجمارك (CUSTOM) ، مصاريف تحميل وتنزيل (LBR_TRANS) ، إجمالي المصاريف (TOTAL) .

إنشاء ملفات الفهرسة

توضع البيانات في ملف قاعدة البيانات بترتيب دخولها أول مرة فإذا أردت أن تبحث في الملف أو أن تظهر بياناته بترتيب مغاير لترتيبها داخل الملف فلا بد من عمل فهرسة (Indexing) للملف قاعدة البيانات.

إذن الغرض من الفهرسة هو إظهار البيانات الموجودة بالملف بترتيب مغاير لترتيب دخولها إلى الملف لتسهيل متابعتها أو البحث فيها. والنظام الذي بين أيدينا يتطلب ترتيب الملف بطريقتين:

أولاً: طبقاً لرقم الاعتماد ويستخدم هذا الرقم لتمييز الاعتمادات عن بعضها في حالة إضافة اعتماد جديد أو تسجيل مصروفات اعتماد أو إظهار محتوياته.
ثانياً: طبقاً لتاريخ الاعتماد وذلك لتسهيل استخراج التقارير المطلوبة عن اعتمادات فترة زمنية معينة أو قبل أو بعد تاريخ معين.

ويوضح شكل ٢ - ٤ أسماء ملفات الفهرسة والحقول التابعة لها والمستخدمه في النظام.

ملف قاعدة البيانات Database file	حقل الفهرسة Key	اسم ملف الفهرس Index file	طريقة ترتيب الملف Indexed By
ORDER	ORD_NO ORD_DAT	ORDER.NDX IDATE.NDX	طبقاً لرقم الاعتماد طبقاً لتاريخ الاعتماد

شكل ٢ - ٤ أسماء ملفات الفهرسة والحقول التابعة لها

ونوضح أن ملف الفهرس (Index file) يجوز أن ينشأ حتى لو كان ملف قاعدة البيانات خالياً من البيانات. ولذلك ننصح بإنشاء ملفات الفهرسة بعد الانتهاء من إنشاء ملف قاعدة البيانات لتكون جاهزة لأننا سنعتمد عليها في كل برامج النظام.

ولا توجد قاعدة ننصح بها عند اختيار ترتيب الملف بل يعتمد ذلك على حاجة المبرمج . وعلى التصميم الذي اختاره للنظام .

انشئ ملفات الفهرسة بعد الانتهاء من إدخال مواصفات ملف قاعدة البيانات (ويجب أن يكون ملف قاعدة البيانات مفتوحاً) بالأمرين التاليين :

. INDEX ON ORD_NO TO ORDER

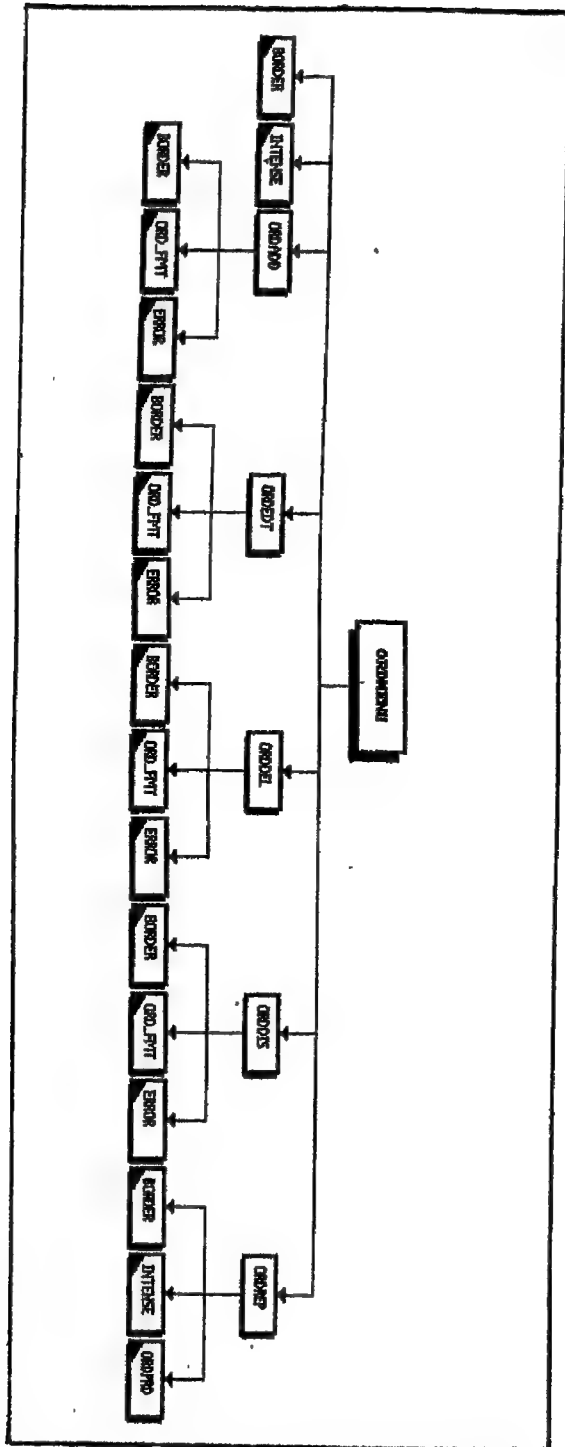
. INDEX ON ORD_DAT TO IDATE

ملاحظة : سنشرح طريقة عملية أخرى للفهرسة عند شرح نظام صيانة الملفات في الفصل التاسع .

تحديد برامج النظام

الخطوة التالية في تطوير النظام هي رسم خريطة للنظام توضح البرامج التي يشتمل عليها ويجب أن تشتمل هذه الخريطة على الوظائف المطلوبة من النظام . وهذه الخريطة هي التي سنعتمد عليها في تطوير البرامج بعد ذلك .

ويوضح شكل ٣ - ٤ خريطة النظام التي توضح العلاقة بين البرامج التي يشتمل عليها . وفي هذه الخريطة تم تقسيم النظام إلى وظائف أساسية هي : الإضافة (Add) والتعديل (Edit) والحذف (Delete) والإظهار (View) والتقارير (Reports) ويمثل كل وظيفة من هذه الوظائف مستطيل داخل خريطة النظام وكلها في مستوى واحد . ولأننا نستخدم طريقة البرمجة التركيبية (Structured programming) فإننا سنعد لكل وظيفة من هذه الوظائف برنامجاً خاصاً بها أما البرامج الصغيرة التي يستخدمها أكثر من برنامج وتسمى Procedures أو Submodules فتجدها في ملف الإجراءات وتعرفها في الخريطة بتظليل الركن الجنوبي الأيسر من المستطيل الذي يمثلها . وستتناول فيما يلي شرح البرامج التي تؤدي هذه الوظائف وشرح البرامج المتفرعة عنها والتي تلزم لأداء غرض محدد داخل البرنامج . بالإضافة إلى الإجراءات الجديدة والتي لم نتعرض لها في الفصل السابق .



شكل ٣ - ٤ خريطة نظام الاعتمادات

قائمة نظام الاعتمادات المستندية

يجب أن تلبي قائمة الاعتمادات (أوامر الشراء) الوظائف الأساسية الموضحة بخريطة النظام وهي : إضافة اعتماد جديد وتعديل بيانات اعتماد وحذف اعتماد وإظهار محتويات اعتماد والتقارير. ويوضح شكل ٤ - ٤ القائمة الرئيسية لنظام

Tuesday 26, February 1991 Time: 14:55:44

Arabian Computer Center
C.R. 39387

Purchase orders menu

- (1) Add a new order (s).....
- (2) Edit an existing order (s).....
- (3) Delete an order (s).....
- (4) View an order (s).....
- (5) Reports.....
- (6) Go to main menu.....
- (Q) Quit to DOS.....

Select number or colored letter ==> █

شكل ٤ - ٤ القائمة الرئيسية لنظام الاعتمادات

الاعتمادات والاختيارات التي تشتمل عليها. كما يشتمل شكل ٥ - ٤ على برنامج ORDMENU.PRG اللازم لإظهار هذه القائمة وتنفيذ اختياراتها. وتشبه فكرة هذا البرنامج والأوامر التي يشتمل عليها برنامج DBAPP.PRG الذي تقدم في الفصل السابق ويستخدم نفس الإجراءات التي شرحناها كذلك. والفرق بين هذا البرنامج وبرنامج DBAPP.PRG أننا استخدمنا هنا علامة \$ للدلالة على وجود سلسلة من الحروف فمثلاً هذا الأمر

CASE choice \$ "1Aa"

```

1 * -----*
2 * Program.....: ORDMENU.PRG*
3 * Called form.: DBAPP.PRG*
4 * Programmer...: Magdi M. Abu Al-Ata*
5 * Date.....: Dec. 25,1990*
6 * Purpose.....: To display purchase orders menu*
7 * -----*
8 PRIVATE scrttitle,scrfootr,choice
9 scrttitle = "purchase orders menu"
10 scrfootr = "Select number or colored letter ==>"
11 ok = .T.
12 DO WHILE ok
13     choice = " "
14     DO border WITH scrttitle,scrfootr
15     * Draw menu
16     @10, 5 SAY "(1) Add a new order (s)....."
17     DO intense WITH 10,10,"A"
18     @11, 5 SAY "(2) Edit an existing order (s)"
19     DO intense WITH 11,10,"E"
20     @12, 5 SAY "(3) Delete an order (s)....."
21     DO intense WITH 12,10,"D"
22     @13, 5 SAY "(4) View an order (s)....."
23     DO intense WITH 13,10,"V"
24     @14, 5 SAY "(5) Reports....."
25     DO intense WITH 14,10,"R"
26     @15, 5 SAY "(G) Go to main menu....."
27     DO intense WITH 15,10,"G"
28     @16, 5 SAY "(Q) Quit to DOS....."
29     DO intense WITH 16,10,"Q"
30     @22,43 GET choice
31     READ
32     * Do selected option
33     DO CASE
34     CASE choice $ "1Aa"
35         DO ordadd                                && Add new order
36     CASE choice $ "2Ee"
37         DO ordedt                                && Edit an order
38     CASE choice $ "3Dd"
39         DO ordel                                  && Delete an order
40     CASE choice $ "4Vv"
41         DO orddis                                && Display an order
42     CASE choice $ "5Rr"
43         DO ordrep                                && Dispaly reports menu
44     CASE choice $ "Gg"
45         RETURN                                    && Go to Main menu
46     CASE choice $ "Qq"
47         CLEAR
48         QUIT                                    && Quit to DOS
49     OTHERWISE
50         LOOP
51     ENDCASE
52 ENDD
53 *****
54 * End of file ORDMENU.PRG *
55 *****

```

معناه إذا كانت قيمة choice 1 أو A أو a وهو مساوٍ للأمر

CASE choice = "1".OR. UPPER (choice) = "A"

ومعناه أيضاً إذا كانت قيمة choice 1 أو إذا كانت القيمة بعد تحويلها إلى حرف كبير (Upper) تساوي A .

إضافة اعتماد جديد إلى الملف

لو رجعنا إلى قائمة النظام في شكل ٤ - ٤ ستجد أول اختيار في القائمة هو إضافة اعتماد جديد للملف (Add a new order) . وتبدأ أول خطوة في هذا النظام وأي نظام آخر بإدخال البيانات .

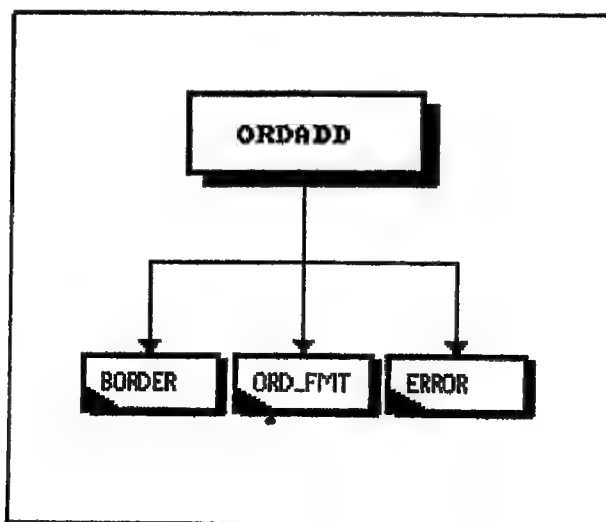
ورغم أننا يمكننا إدخال سجلات جديدة بأمر APPEND واستخدام الشاشة المجهزة من قبل الشركة المنتجة لقاعدة البيانات إلا أننا لجأنا إلى إعداد برنامج مستقل لأن هذا البرنامج يظهر شاشة منظمة لإدخال البيانات (انظر شكل ٦ - ٤) ويتحكم في عدم تكرار السجلات الداخلة إلى الملف .

Tuesday 26, February 1991		Time: 14:58:56	
Arabian Computer Center C.R. 39387			
Order main data			
Order #		Order value	0.00
Order kind (FOB/CAD/CAF)		Order date	
Order description		Shipping name	
Company name		Company address	
Bank name		Bank address	
Bill of lading #		Document shipping #	
Commission	0.00	Insurance	0.00
Labor transportation		Postage	0.00
Total		Customs	0.00
Adding a new order			
Press Enter to return to the menu			

شكل ٦ - ٤ شاشة إدخال بيانات الاعتماد

البرنامج اللازم لإضافة اعتماد جديد إلى الملف هو ORDADD.PRG وتتلخص وظيفته في إدخال بيانات الاعتماد بعد التأكد أن رقم الاعتماد غير موجود من قبل بالملف ويوضح شكل ٦ - ٤ شاشة إدخال بيانات الاعتماد ومنه تلاحظ أن شاشة إدخال بيانات الاعتماد مقسومة قسمين والغاية من ذلك وضع بيانات الاعتماد في الجزء العلوي من الشاشة. لأن هذه البيانات هي التي تخص الاعتماد عند فتحه أما القسم السفلي من الشاشة فيخصص مصروفات الاعتماد ولأن المصروفات لا تعرف قبل وصول الاعتماد فقد لجأنا إلى تعبئة هذا الجزء في شاشة التعديل أي عندما يصل الاعتماد وتعرف مصروفاته.

ويشمل شكل ٧ - ٤ على الخريطة التي توضح علاقة هذا البرنامج بالبرامج الأخرى داخل النظام. كما يشتمل شكل ٨ - ٤ على محتويات هذا البرنامج.



شكل ٧ - ٤ خريطة برنامج ORDADD.PRG

```

1 * -----*
2 * Program   : ORDADD.PRG                      *
3 * Author    : Magdi M. Abu Al-Ata             *
4 * Date      : October 24, 1990                 *
5 * Purpose   : Adds new order                  *
6 * Called from: ORDMENU.PRG                     *
7 * Called from: DPAPP.PRG                       *
8 * -----*
9 USE order INDEX order      && Indexed on ord_no
10 DO WHILE .T.
11     DO border WITH "Order main data", "Adding a new order"
12     DO ord_fmt
13     @24,0
14     @ 24,23 SAY "Press Enter to return to the menu"
15     @ 10,28 GET m_ord_no PICTURE "@!"
16     READ
17     SEEK m_ord_no
18     DO CASE
19     CASE m_ord_no = SPACE(5)
    
```

شكل ٨ - ٤ برنامج ORDADD.PRG

```

20 <-----EXIT
21 CASE FOUND()
22 | DO error WITH "Order No. " + TRIM(m_ord_no) + " already exist."
23 -----LOOP
24 CASE .NOT. FOUND()
25 | APPEND BLANK
26 | @ 24,0
27 | REPLACE ord_no with m_ord_no
28 ENDCASE
29 @ 24,0
30 @ 10, 64 GET m_o_val PICTURE "9,999.99"
31 @ 11, 28 GET m_o_knd PICTURE "a!"
32 @ 11, 64 GET m_o_dat
33 @ 12, 28 GET m_o_des
34 @ 12, 64 GET m_sh_nm
35 @ 13, 28 GET m_co_nm
36 @ 13, 64 GET m_co_ad
37 @ 14, 28 GET m_b_nm
38 @ 14, 64 GET m_b_ad
39 @ 15, 28 GET m_bl_no
40 @ 15, 64 GET m_doc_no
41 @ 9, 2 TO 16, 77
42 @ 17, 2 TO 21, 77
43 READ
44 STORE " " TO m_ok
45 @ 24, 24 SAY "Valid order data? (Y/N)---) "GET m_ok
46 READ
47 IF UPPER (m_ok) = "Y"
48 | @ 24,0
49 | REPLACE ord_val with m_o_val, ord_kind with m_o_knd, ;
49 | ord_dat with m_o_dat, ord_desc with m_o_des, ;
49 | ship_nm with m_sh_nm
50 | REPLACE comp_nm with m_co_nm, comp_adr with m_co_ad, ;
50 | bnk_nm with m_b_nm, bnk_adr with m_b_ad, ;
50 | bill_no with m_bl_no, doc_no with m_doc_no
51 ELSE
52 | DELETE && Delete the APPENDED record
53 ENDIF
54 ENDDO
55 CLOSE DATABASES
56
57 *****
58 * End of file ORDADD.PRG *
59 *****

```

تابع شكل ٨ - ٤ برنامج ORDADD.PRG

وعن هذا البرنامج نوضح ما يلي:

- يبدأ البرنامج بفتح ملف قاعدة البيانات مرتباً طبقاً لأرقام الاعتمادات (سطر رقم ٩).

- بعد ذلك يفتح دواة (١٠ - ٥٤) لتكرار إدخال الاعتمادات حتى يدخل المستفيد

- فراغات بدلاً من رقم الحساب (أي يضغط مفتاح الإدخال للخروج من الدوارة وبالتالي البرنامج).
- تبدأ الدوارة برسم البرواز الخارجي للشاشة (سطر رقم ١١) ويقوم بهذه المهمة الإجراء `border` (شرحناه في الفصل السابق). ثم استدعاء الإجراء `(ord_fmt)` ومهمته إظهار شاشة منظمة لإدخال سجل جديد إلى الملف (راجع شكل ٦ - ٤) وسوف نناقش هذا الإجراء بعد الانتهاء من شرح البرنامج.
- تظهر دائماً رسالة في سطر ٢٤ قبل إدخال رقم الاعتماد والسؤال عنه لتوضيح للمستفيد أن الرجوع للقائمة يتم بضغط مفتاح الإدخال (سطر رقم ١٤).
- يقبل البرنامج رقم الاعتماد (حتى ٥ حروف) من المستفيد (سطر ١٥ - ٦٦) ويضع هذا الرقم في حقل الذاكرة `m_ord_no`. وهذا الحقل أنشأه الإجراء `ord_fmt` الذي استدعي في أول البرنامج.
- بعد إدخال رقم الاعتماد من لوحة المفاتيح يبدأ البرنامج في البحث عن هذا الرقم في الملف ثم يوجّه لتنفيذ واحدة من الحالات التالية (سطر رقم ١٨).
- الحالة الأولى : إدخال فراغات محل رقم الحساب وهي تسبب الخروج من الدوارة وبالتالي الرجوع إلى القائمة (سطر ١٩ - ٢٠).
- الحالة الثانية : أن يكون الرقم موجوداً من قبل. وهنا يظهر البرنامج رسالة تحذيرية تفيد أن هذا الرقم موجود من قبل وهذه الرسالة تظهر نتيجة استدعاء الإجراء `error` (أيضاً سنناقشه بعد الانتهاء من البرنامج).
- ثم يعيد التنفيذ مرة أخرى إلى بداية الدوارة ليتاح للمستفيد إدخال الرقم الصحيح (سطر ٢١ - ٢٣).
- الحالة الثالثة : أن يكون الرقم جديداً وغير موجود في الملف (سطر رقم ٢٤) وهنا يضاف سجل خالٍ في نهاية الملف ثم يستبدل رقم الاعتماد بالرقم الذي أدخل من لوحة المفاتيح (سطر ٢٤ - ٢٧).
- بعد ذلك يسمح البرنامج بإدخال باقي بيانات العميل (سطر ٣٠ - ٤٣). وهنا يجب أن نوضح أن المعلومات التي تدخل من لوحة المفاتيح تسجل أولاً بالذاكرة لأن الكتابة في السطور من ٣٠ - ٤٢ تتم في حقول ذاكرة (كلها تبدأ بحرف `m_`

- تمهيداً لنقلها في الملف بعد ذلك إذا تأكد المستفيد من صحتها).
- بعد إدخال البيانات تظهر رسالة مفادها هل بيانات هذا العميل صحيحة؟ ويتم الرد عليها إما بحرف Y أو N.
- بعد الرد على الرسالة يتم التحقق من الإجابة هل هي Y (نعم) أم غير ذلك (سطر ٤٧ - ٥٣) فإذا كانت الإجابة Y أي نعم فمعنى ذلك أن البيانات صحيحة وبالتالي تستبدل حقول الملف بحقول الذاكرة التي تشتمل على البيانات التي أدخلت. ولا يحذف السجل الذي أضيف في السطر رقم ٢٥ لأن بياناته خاطئة.
- بعد الانتهاء من إدخال البيانات والرغبة في الخروج يغلق ملف قاعدة البيانات (سطر رقم ٥٥).

والآن سنناقش الإجراءات الواردة بهذا البرنامج.

الإجراء `ord_fmt`

- هذا الإجراء وظيفته رسم شاشة إدخال بيانات منظمة بشكل مرضٍ يدفع الملل عن مدخل البيانات. ويشتمل شكل ٩ - ٤ على هذا الإجراء ويبدأ بإعلان حقول الذاكرة التي سينشأها عامة لكي تستخدم مع باقي البرامج بنفس الاسم (سطر ١١٩). ثم يضع فراغات بحقول بالذاكرة تقابل الحقول الموجودة بالملف وتأخذ نفس الطول. ولسهولة التعرف على حقل الذاكرة المقابل لحقل الملف وحدنا الاسم تقريباً غير أن اسم حقل الذاكرة يبدأ بحرفي M (سطور ١٢١ - ١٢٥).
- السطور من ١٢٦ - ١٦٢ لرسم شاشة إدخال البيانات. وهذه الشاشة واحدة مع باقي برامج التعديل والحذف والاسترجاع وكلها تستخدم أمر `GET...SAY...@` ولذلك يمكن الاستفادة من مصمم الشاشات الموجود بقاعدة البيانات لرسم الشاشة بالشكل المطلوب وذلك باستخدام أمر:
`CREATE SCREEN ord_fmt`
- ثم تعديل ملف `ord_fmt.FMT` بإضافة الأوامر الأخرى (سنناقش بعد قليل كيفية تصميم هذا الملف).

- أمر CLEAR GETS في سطر ١٦٣ يمنع صلاحيات التعديل في الحقول التي تتاح نتيجة أمر @...GET بعبارة أخرى تمنع مؤشر الشاشة من الوقوف عند الحقول التي تستدعي بأمر @...GET ولذلك فإن الفراغات التي خزناها في حقول الذاكرة مساوية لمساحة الحقول ستظهر أمام أسماء الحقول لتحدد أين ستدخل البيانات. ولذلك فإننا نستخدم أمر @...GET داخل برنامج الإضافة ORDADD.PRG مرة ثانية بعد رسم هذه الشاشة لإدخال بيانات هذه الحقول.

ونوضح فيما يلي طريقة سهلة للحصول على شكل ٩ - ٤ بدلاً من كتابة الأوامر أمراً أمراً وتصحيحها في حالة حدوث خطأ أثناء التجارب :

```
114 *****
115 * Procedure : ord_fmt
116 * Purpose   : Drows order screen format
117 *****

118 PROCEDURE ord_fmt
119 PUBLIC m_ord_no, m_o_val, m_o_knd, m_o_dat, m_o_des, m_sh_rm, ;
119 m_co_rm, m_co_ad, m_b_rm, m_b_ad, m_bl_no, m_doc_no, m_comm, ;
119 m_post, m_ins, m_custom, m_lbr, m_total
120 * Give those variables their starting values
121 STORE SPACE(5) TO m_ord_no, m_bl_no, m_doc_no
122 STORE 0 TO m_o_val, m_comm, m_post, m_ins, m_custom, m_lbr, m_total
123 STORE SPACE(3) TO m_o_knd
124 STORE SPACE(12) TO m_o_des, m_sh_rm, m_co_rm, m_co_ad, m_b_rm, m_b_ad
125 STORE CTOD(" / / - ") TO m_o_dat
126 @ 10, 4 SAY "Order #"
127 @ 10, 28 SAY m_ord_no
128 @ 10, 42 SAY "Order value"
129 @ 10, 64 GET m_o_val PICTURE "9,999.99"
130 @ 11, 4 SAY "Order kind(FOB/CAD/CAF)"
131 @ 11, 28 GET m_o_knd
132 @ 11, 42 SAY "Order date"
133 @ 11, 64 GET m_o_dat
134 @ 12, 4 SAY "Order description"
135 @ 12, 28 GET m_o_des
136 @ 12, 42 SAY "Shipping name"
137 @ 12, 64 GET m_sh_rm
138 @ 13, 4 SAY "Company name"
139 @ 13, 28 GET m_co_rm
140 @ 13, 42 SAY "Company address"
141 @ 13, 64 GET m_co_ad
```

شكل ٩ - ٤ الإجراء ord_fmt

```

142 @ 14, 4 SAY "Bank name"
143 @ 14, 28 GET m_b_nm
144 @ 14, 42 SAY "Bank address"
145 @ 14, 64 GET m_b_ad
146 @ 15, 4 SAY "Bill of loading #"
147 @ 15, 28 GET m_bl_no
148 @ 15, 42 SAY "Document shipping #"
149 @ 15, 64 GET m_doc_no
150 @ 18, 5 SAY "Commision"
151 @ 18, 17 GET m_comm PICTURE "9,999.99"
152 @ 18, 28 SAY "Insurance"
153 @ 18, 38 GET m_ins PICTURE "9,999.99"
154 @ 18, 51 SAY "Postage"
155 @ 18, 65 GET m_post PICTURE "9,999.99"
156 @ 19, 5 SAY "Labor transportation"
157 @ 19, 38 GET m_lbr PICTURE "9,999.99"
158 @ 19, 51 SAY "Customs"
159 @ 19, 65 GET m_custom PICTURE "9,999.99"
160 @ 20, 5 SAY "Total"
161 @ 9, 2 TO 16, 77
162 @ 17, 2 TO 21, 77
163 CLEAR GETS
164 RETURN
165

```

تابع شكل ٩ - ٤

١ - من نقطة توجيه الأوامر أدخل الأمر الآتي بعد فتح ملف ORDER.DBF

CREATE SCREEN ORD_FMT

ستحصل على شاشة تستخدم لتصميم شاشة إدخال البيانات (انظر شكل ١٠ - ٤).

٢ - اختر Load Fields تحت Set Up (انظر شكل ١٠ - ٤) ثم اختر جميع الحقول التي ستظهر بعد ذلك (الاختيار يتم بإضاءة الحقل ثم ضغط مفتاح الإدخال). وعندما تظهر لك سبورة تشتمل على جميع الحقول التي اخترتها استخدم مفاهيم إدخال مسافات أو سطور خالية وتحريك الحقول من أماكنها وإعادة كتابة أسمائها ورسم إطار فردي أو مزدوج داخل الشاشة للحصول على الشكل المقترح لشاشة إدخال البيانات الموجود في شكل ١١ - ٤. ثم احفظ هذا الشكل. (استخدم المفاهيم الواردة بكتابنا المرجع الأساسي لقاعدة البيانات dBASE III PLUS في الفصل الرابع من الجزء الأول تحت عنوان إعدادات شاشة إدخال لتوافق

Set Up Modify Options Exit 19:20:12 pm

Select Database File
Create New Database File
Load Fields

CREATE SCREEN <<C> ORD_FMT.SCR Opt: 3/3

Position selection bar - F1. Select - <F1>. Leave menu - <F2>. Blackboard - F10.
Select fields to load onto the blackboard.

شكل ١٠ - ٤ تصميم شاشة الإدخال

Set Up Modify Options Exit 19:26:02 pm

Order #	XXXXX	Order value	9999.99
Order kind(FOB/CAD/CAF)	XXX	Order date	99/99/99
Order description	XXXXXXXXXXXX	Shipping name	XXXXXXXXXXXX
Company name	XXXXXXXXXXXX	Company address	XXXXXXXXXXXX
Bank name	XXXXXXXXXXXX	Bank address	XXXXXXXXXXXX
Bill of loading #	XXXXX	Document shipping #	99999

Commission	9999.99	Insurance	9999.99	Postage	9999.99
Labor transportation	9999.99		9999.99	Customs	99999.99
Total	9999.99				

MODIFY SCREEN <<C> ORD_FMT.SCR Pg 01 Row 22 Col 00

Enter text. Drag field or box under cursor with <F1>... F10 for menu.
Screen field definition blackboard

شكل ١١ - ٤ شكل شاشة الإدخال

استخدامات المستفيد إذا لم تكن تعرف كيفية إعداد هذا الشكل).

٣ - باتباع الخطوتين السابقتين أصبح عندك شاشة جاهزة. تستخدم لأغراض إدخال أو تعديل أو إظهار أو حتى حذف بيانات الاعتماد ونوضح هنا أن قاعدة البيانات تنشئ ملفين اثنين نتيجة حفظ الشاشة السابقة هما:

ORD_FMT.FMT - ٢

ORD_FMT.SCR - ١

الأول يشتمل على الشكل المقترح لشاشة إدخال البيانات والثاني يشتمل على الأوامر اللازمة لإظهار هذه الشاشة وهذا الأخير يمكن تعديل أوامره بواسطة أي منسق للنصوص أو محرر للسطور معروف لك أو حتى محرر السطور الموجود داخل قاعدة البيانات لأنه مكتوب بشفرة ASCII. ويشتمل شكل ١٢ - ٤ على محتويات ملف ORD_FMT.FMT قبل تعديلها والتي تم كتابتها بواسطة قاعدة البيانات.

```

@ 10, 4 SAY "Order #"
@ 10, 28 GET ORDER->ORD_NO
@ 10, 42 SAY "Order value"
@ 10, 64 GET ORDER->ORD_VAL
@ 11, 4 SAY "Order kind(FOB/CAD/CAF)"
@ 11, 28 GET ORDER->ORD_KIND
@ 11, 42 SAY "Order date"
@ 11, 64 GET ORDER->ORD_DAT
@ 12, 4 SAY "Order description"
@ 12, 28 GET ORDER->ORD_DESC
@ 12, 42 SAY "Shipping name"
@ 12, 64 GET ORDER->SHIP_NAM
@ 13, 4 SAY "Company name"
@ 13, 28 GET ORDER->COMP_NAM
@ 13, 42 SAY "Company address"
@ 13, 64 GET ORDER->COMP_ADR
@ 14, 4 SAY "Bank name"
@ 14, 28 GET ORDER->BNK_NAM
@ 14, 42 SAY "Bank address"
@ 14, 64 GET ORDER->BNK_ADR
@ 15, 4 SAY "Bill of loading #"
@ 15, 28 GET ORDER->BILL_NO
@ 15, 42 SAY "Document shipping #"
@ 15, 64 GET ORDER->DOC_NO
@ 18, 5 SAY "Comission"
@ 18, 17 GET ORDER->COMISION
@ 18, 28 SAY "Insurance"

```

شكل ١٢ - ٤ محتويات ملف ORDFMT.FMT قبل تعديلها

```
@ 18, 38 GET ORDER->GOOD_INS
@ 18, 51 SAY "Postage"
@ 18, 65 GET ORDER->POSTAGE
@ 19, 5 SAY "Labor transportation"
@ 19, 38 GET ORDER->LBR_TRNS
@ 19, 51 SAY "Customs"
@ 19, 65 GET ORDER->CUSTOM PICTURE "99999.99"
@ 20, 5 SAY "Total"
@ 20, 17 GET ORDER->TOTAL
@ 9, 2 TO 16, 77
@ 17, 2 TO 21, 77
```

تابع شكل ١٢ - ٤

٤ - استخدم محرر السطور المألوف لك لتعديل شكل ١٢ - ٤ لكي يصبح موافقاً لشكل ٩ - ٤ ويتلخص التعديل الذي تم على شكل ١٢ - ٤ لكي يصبح هو نفسه شكل ٩ - ٤ في الآتي:

* إضافة أوامر STORE بعد ذلك لإنشاء حقول بالذاكرة (Memory variables) لتقابل الحقول الموجودة بالملف الأصلي والتي ستظهر في شاشة إدخال البيانات وتلاحظ أن الحقول الحرفية تنشأ بفراغات (spaces) مساوية لعدد فراغات الحقل الأصلي والرقمية بتخزين الرقم صفر والتاريخية باستخدام الوظيفة . CTOD (" / / ")

* إضافة أمر PUBLIC في أول الإجراء ومهمته إعلان حقول الذاكرة عامة لتكون حقول هذا الإجراء متاحة مع البرامج الأخرى التي تستخدمه مثل برامج التعديل أو الإظهار.

* إضافة أمر CLEAR GETS لإلغاء صلاحيات الإدخال أو التعديل التي تتيحها أوامر @...GET لأننا نستخدم هذا الإجراء لرسم الشاشة فقط . وفي حالة الحاجة لإدخال أو تعديل بيانات الحقول فإننا نستخدم أوامر @...GET مرة ثانية .

الإجراء error

أيضاً هذا الإجراء نستخدمه معظم برامج النظام ومهمته إظهار رسالة خطأ في حالة حدوث خطأ ما ويشتمل شكل ١٣ - ٤ على هذا الإجراء ومنه تلاحظ:

```

97 *****
98 * Procedure : error
99 * Purpose   : To display an error message and pause execution
100 *****

101 PROCEDURE error
102 PARAMETER errmsg
103 IF mcolor
104 |   SET COLOR TO R*/N
105 ELSE
106 |   SET COLOR TO W*/N
107 ENDIF
108 @ 24,1 SAY  errmsg + msg
109 READ
110 @ 24,1
111 SET COLOR TO W*/N
112 RETURN
113

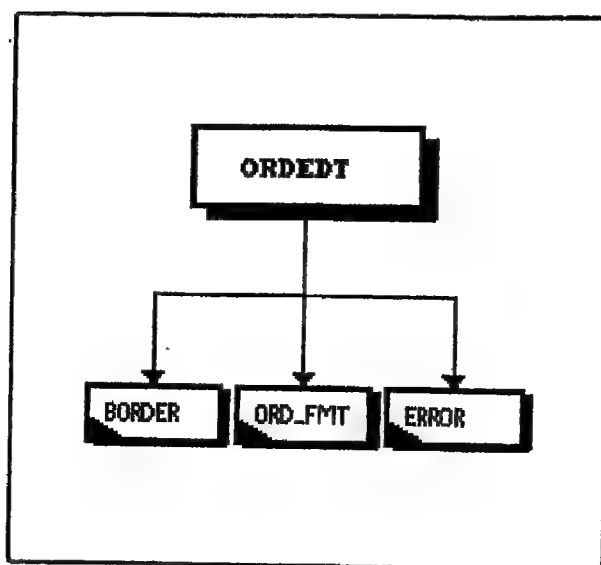
```

شكل ١٣ - ٤ الإجراء error

- أن الإجراء يقبل معطيات من خارجه عبارة عن الرسالة التي ستظهر للمستخدم (سطر ١٠٢).
- يبدأ الإجراء بالتحقق من نوع الشاشة هل هي ملونة أم لا (سطر ١٠٣ - ١٠٧).
- فإذا كانت ملونة (سطر ١٠٤) استخدم ألواناً أخرى لتظهر الرسالة بلون مخالف لشد انتباه المستخدم (سطر ١٠٤). وإلا (سطر ١٠٥) استخدم الألوان المختارة (سطر ١٠٦).
- يظهر رسالة الخطأ على الشاشة في سطر ٢٤ بالألوان التي اختيرت (سطر ١٠٨).
- وتلاحظ أن الرسالة التي ستظهر هي الرسالة التي ستدخل للإجراء من برنامج آخر بالإضافة إلى العبارة المخزنة بحقل الذاكرة msg .
- أمر READ في سطر ١٠٩ مهمته إيقاف البرنامج مؤقتاً ليتمكن المستخدم من متابعة رسالة الخطأ التي تظهر. وبمجرد ضغط مفتاح الإدخال يستأنف البرنامج العمل من محل توقفه.
- بعد ذلك تحذف الرسالة (سطر ١١٠) وتعاد الألوان إلى الألوان المختارة للكتابة (سطر ١١١).

تعديل بيانات اعتماد

نحتاج لتعديل بيانات الاعتماد لتصحيح خطأ أثناء إدخال بيانات الاعتماد أو لتسجيل المصروفات التي تركناها عندما أضفنا الاعتماد أول مرة ويتم تسجيل المصروفات حال وصول الاعتماد ومعرفة هذه المصروفات. ولذلك فإن برنامج التعديل يستخدم نفس شاشة تسجيل بيانات الاعتماد أول مرة. والبرنامج الذي يقوم بهذه المهمة هو ORDED.T.PRG ويشتمل شكل ١٤ - ٤ على الخريطة التي توضح



شكل ١٤ - ٤ خريطة برنامج ORDED.T.PRG

علاقة هذا البرنامج بالبرامج الأخرى داخل النظام كما يشتمل شكل ١٥ - ٤ على محتويات هذا البرنامج. وتتلخص طريقة عمله في التأكد أولاً أن رقم الاعتماد المطلوب تعديله أو تسجيل مصروفاته موجود بملف الاعتمادات (ORDER.DBF). فإذا كان رقم الاعتماد موجوداً فيتم إحضار الشاشة المعدة لهذا الغرض (راجع شكل ٦ - ٤). وتلاحظ أن صلاحيات التعديل في هذا البرنامج لا تقتصر فقط على بيانات الاعتماد الأساسية داخل المستطيل العلوي من الشاشة وإنما تتعداها إلى بيانات

```

1 * -----*
2 * Program      : ORDEDT.PRG                      *
3 * Author       : Magdi M. Abu Al-Ata            *
4 * Date         : October 24, 1990                *
5 * Purpose      : To edit an existing order       *
6 * Called from:  ORDMENU.PRG                      *
7 * Called from:  DBAPP.PRG                        *
8 * -----*
9 USE order INDEX order
10 DO WHILE .T.
11     DO border WITH "Order main data", "Modifying order data"
12     DO ord_fmt
13     @24,0
14     @ 24,24 SAY "Press Enter to return to the menu"
15     @ 10,28 GET m_ord_no
16     READ
17     @24,0
18     SEEK m_ord_no
19     DO CASE
20     CASE m_ord_no = SPACE(5)
21     <-----EXIT
22     CASE .NOT. FOUND()
23     DO error WITH "Order No. " + TRIM(m_ord_no) + " Not found."
24     -----LOOP
25     CASE FOUND()
26     @ 24,0
27     @ 10, 64 GET ORD_VAL PICTURE "9,999.99"
28     @ 11, 28 GET ORD_KIND
29     @ 11, 64 GET ORD_DAT
30     @ 12, 28 GET ORD_DESC
31     @ 12, 64 GET SHIP_NAM
32     @ 13, 28 GET COMP_NAM
33     @ 13, 64 GET COMP_ADR*
34     @ 14, 28 GET BNK_NAM
35     @ 14, 64 GET BNK_ADR
36     @ 15, 28 GET BILL_NO
37     @ 15, 64 GET DOC_NO
38     @ 18, 17 GET COMISION PICTURE "9,999.99"
39     @ 18, 38 GET GOOD_INS PICTURE "9,999.99"
40     @ 18, 65 GET POSTAGE PICTURE "9,999.99"
41     @ 19, 38 GET LBR_TRNS PICTURE "9,999.99"
42     @ 19, 65 GET CUSTOM PICTURE "9,999.99"
43     @ 9, 2 TO 16, 77
44     @ 17, 2 TO 21, 77
45     READ
46     REPLACE TOTAL WITH COMISION + GOOD_INS + POSTAGE + LBR_TRNS +CUSTOM
47     @ 20, 17 SAY TOTAL PICTURE "9,999.99"
48     ENDCASE
49     * Determine if the operator wants to edit another.
50     STORE " " TO m_ok
51     @ 24, 0
52     @ 24, 20 SAY "Edit another order record? (Y/N)----> " GET m_ok
53     READ
54     IF UPPER (m_ok) = "N"

```

```
55 |<-----EXIT
56 |      ENDIF
57 |      @24,0
58 ENDDO
59 CLOSE DATABASES
60 RETURN
61
62 *****
63 * End of file ORDEDT.PRG *
64 *****
```

تابع شكل ١٥ - ٤ برنامج ORDEDT.PRG

المصروفات الموجودة في المستطيل السفلي من الشاشة. ولذلك فإن أوامر GET...@ في هذا البرنامج تشتمل على حقول المصروفات أيضاً. وبمجرد تسجيل المصروفات فإن حقل TOTAL يتم تعبئته تلقائياً ويظهر المجموع أمامك على الشاشة.

لاحظ الأمر الموجود في سطر رقم ٤٦ ومعناه أن حقل إجمالي المصروفات TOTAL يتم تعديله تلقائياً بعد كل عملية تعديل إذ ربما أن التعديل طرأ على أحد بنود المصروفات وطبعاً حقل TOTAL لا يتم إدخاله أو تعديله من الشاشة والهدف من ذلك أن يقوم الحاسب بدلاً من المستخدم بعملية جمع المصروفات التي أدخلت.

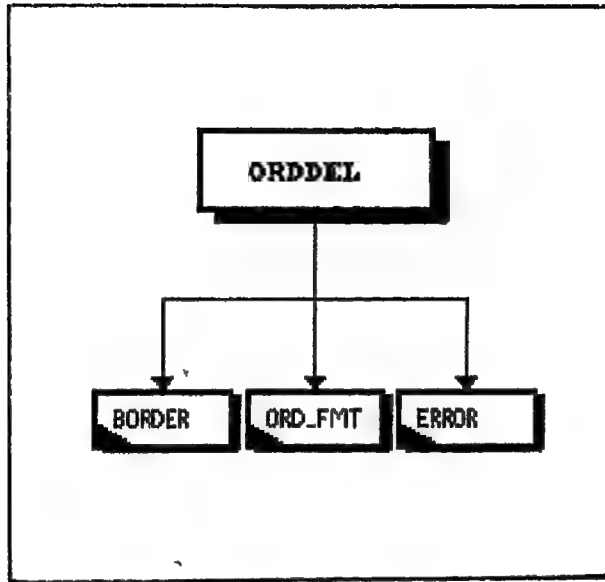
وإليك شرح باقي الأفكار الموجودة بالبرنامج :

- يستدعي البرنامج الإجراء border . مع تغيير الرسالة التي تشرح وظيفة البرنامج (سطر رقم ١١).
- يستدعي البرنامج الإجراء error لإظهار رسالة الخطأ إلا أن الرسالة هنا تفيد أن رقم الاعتماد غير موجود بالملف. وهي عكس الرسالة التي تظهر في حالة الإضافة.
- إذا وجد البرنامج رقم الاعتماد بالملف (سطر رقم ٢٥) فإنه يحضر بيانات هذا الاعتماد مع صلاحيات تعديلها (سطور من ٢٦-٤٥). وتلاحظ أن أوامر GET...@ مشابهة لنظيرتها في برنامج الإضافة إلا أنها هنا تستخدم حقول الملف بدلاً من حقول الذاكرة (انظر شكل ٦ - ٤).

- بعد الانتهاء من تعديل بيانات الاعتماد يسأل البرنامج المستخدم هل تريد تعديل سجل عميل آخر؟ والرد عليها يكون باختيار Y أو N .
- يتم الخروج من الدارة والرجوع للقائمة إذا اختار المستخدم N بمعنى لا ردًا على الرسالة. ومن ثم إغلاق الملف أما إذا اختار Y بمعنى نعم فستكرر الدوارة ليتمكن المستخدم من تعديل بيانات اعتماد آخر.

حذف بيانات اعتماد

الاختيار الثالث في قائمة الاعتمادات حذف اعتماد. والبرنامج الذي يقوم بعملية الحذف هو ORDEL.PRG ويشتمل شكل ١٦ - ٤ على الخريطة التي توضح علاقة هذا البرنامج بالبرامج الأخرى داخل النظام كما يشتمل شكل ١٧ - ٤ على محتويات هذا البرنامج. ويستخدم هذا البرنامج نفس الشاشة المستخدمة لتعديل بيانات اعتماد (راجع شكل ٦ - ٤) مع إلغاء صلاحيات التعديل.



شكل ١٦ - ٤ خريطة برنامج ORDDDEL.PRG

```

1 * -----*
2 * Program   : ORDDDEL.PRG                      *
3 * Author    : Magdi M. Abu Al-Ata             *
4 * Date      : October 24, 1990                 *
5 * Purpose   : To delete an existing order      *
6 * Called from: ORDMENU.PRG                     *
7 * Called from: DPAPP.PRG                       *
8 * -----*
9 USE order INDEX order
10 DO WHILE .T.
11     DO border WITH "Order main data", "Deleting order data"
12     DO ord_fmt
13     @24,0
14     @ 24,24 SAY "Press Enter to return to the menu"
15     @ 10,28 GET m_ord_no
16     READ
17     @24,0
18     SEEK m_ord_no
19     DO CASE
20     CASE m_ord_no = SPACE(5)
21     <-----EXIT
22     CASE .NOT. FOUND()
23     | DO error WITH "Order No. " + TRIM(m_ord_no) + " Not fond."
24     -----LOOP
25     CASE FOUND()
26     @ 24,0
27     @ 10, 64 GET ORD_VAL PICTURE "9,999.99"
28     @ 11, 28 GET ORD_KIND
29     @ 11, 64 GET ORD_DAT
30     @ 12, 28 GET ORD_DESC
31     @ 12, 64 GET SHIP_NAM
32     @ 13, 28 GET COMP_NAM
33     @ 13, 64 GET COMP_ADR
34     @ 14, 28 GET BNK_NAM
35     @ 14, 64 GET BNK_ADR
36     @ 15, 28 GET BILL_NO
37     @ 15, 64 GET DOC_NO
38     @ 18, 17 GET COMISION PICTURE "9,999.99"
39     @ 18, 38 GET GOOD_INS PICTURE "9,999.99"
40     @ 18, 65 GET POSTAGE PICTURE "9,999.99"
41     @ 19, 38 GET LBR_TRNS PICTURE "9,999.99"
42     @ 19, 65 GET CUSTOM PICTURE "9,999.99"
43     @ 9, 2 TO 16, 77
44     @ 17, 2 TO 21, 77
45     CLEAR GETS
46     ENDCASE
47     * Confirm deletion
48     STORE " " TO m_ok
49     @ 24,0
50     @ 24, 21 SAY "Confirm delete this order? (Y/N)-----> " GET m_ok
51     READ
52     IF UPPER (m_ok) = "Y"
53     | DELETE
54     ENDFIF

```

```

55      * Determine if the operator wants to delete another.
56      STORE " " TO yn
57      @ 24, 0
58      @ 24, 21 SAY "Delete another order record? (Y/N)----> " GET yn
59      READ
60      IF UPPER (yn) = "N"
61      <-----EXIT
62      ENDIF
63      @24,0
64 ENDDO
65 CLOSE DATABASES
66 RETURN
67
68 *****
69 * End of file ORDDDEL.PRG *
70 *****

```

تابع شكل ١٧ - ٤ برنامج ORDDDEL.PRG

وتتلخص طريقة عمله في التأكد أولاً من وجود رقم الاعتماد بالملف ثم إحضار بيانات الاعتماد على الشاشة إذا وجده ليتأكد المستفيد أن هذه هي البيانات المطلوب حذفها. تظهر رسالة للتأكيد تسأل العميل هل هذا هو السجل المطلوب للحذف؟ فإذا أجاب نعم علم السجل لغرض الحذف وإلا بقي السجل على ما هو عليه.

وهذا البرنامج شبيه ببرنامج التعديل السابق ويستخدم نفس الشاشة ونفس الملف والإجراءات. ونوضح فيما يلي أوجه الاختلاف بينها والأفكار الجديدة في هذا البرنامج.

- إذا وجد البرنامج الرقم المطلوب داخل الملف فإنه يستخدم نفس أوامر GET...@ لإظهار بيانات العميل (سطر ٢٧ - ٤٤) والاختلاف بين هذا البرنامج وبرنامج التعديل أننا هنا نستخدم أمر CLEAR GETS بعد إحضار حقول السجل لإلغاء صلاحيات تعديلها (سطر رقم ٤٥) أما في برنامج ORDEDIT.PRG فإننا نستخدم أمر READ لإيقاف تنفيذ البرنامج ووضع المؤشر عند أول حقل في مجموعة أوامر GET...@ لتعديلها.

- بعد إظهار بيانات الاعتماد على الشاشة تظهر رسالة للعميل للتأكيد قبل حذف السجل وتطلب منه الرد بنعم (Y) أو لا (N) (سطر ٥٠ - ٥١).

- إذا قرر المستفيد إلغاء بيانات هذا الاعتماد باختيار Y ردًا على الرسالة السابقة فإنه يتم حذف السجل وإلا فإن السجل يبقى على ما هو عليه (سطر ٥٢ - ٥٤).
وننصح بعدم استخدام أمر PACK بعد أمر DELETE في مثل هذه الأحوال لسببين:

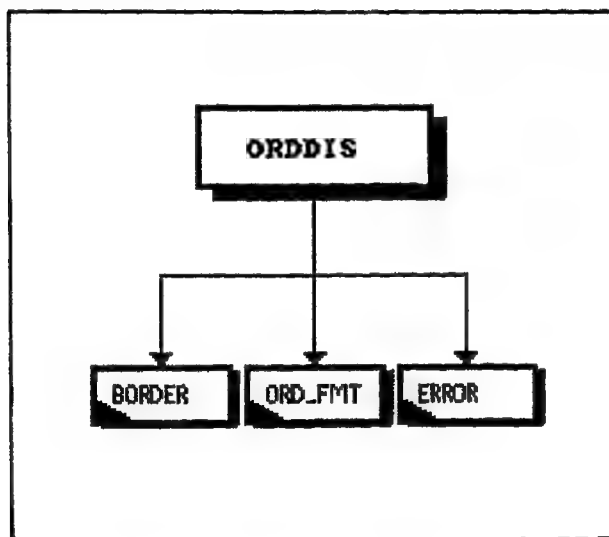
أولاً: لتبقى لدينا إمكانية استرجاع السجل إذا دعت حاجة لذلك.
ثانياً: أن استخدام أمر PACK يسبب تأخير في التنفيذ. ولذلك فمن المناسب تنفيذه ضمن برامج صيانة الملفات (وقد استخدمنا برنامجاً مستقلاً ليتولى عملية حذف السجلات المعلمة بغرض الحذف ضمن برامج صيانة الملفات سنشرحه في الفصل التاسع). إلا أن هذه الطريقة تستلزم أن يكون أمر SET DELETED في حالة ON لكي تتجاهل قاعدة البيانات السجلات المعلمة لأغراض الحذف أثناء البحث في الملف وتعتبرها كأنها غير موجودة. وسبق أن أوضحنا أن هذا الأمر موجود ضمن ملف الإجراءات الذي ينفذ عند بداية تشغيل النظام.

- بعد تعليم السجل لغرض الحذف تظهر رسالة أخرى للمستفيد تسأله ما إذا كان يريد حذف سجل آخر أم لا (سطر ٥٦ - ٥٩).

- إذا أجاب المستفيد على الرسالة بنعم (Y) تتكرر الدوارة الرئيسية في البرنامج لإعادة حذف اعتماد آخر بنفس الطريقة أما إذا أجاب بلا (N) فإن التنفيذ ينتقل للأمر التالي للدوارة الرئيسية (سطر ٦٠ - ٦٢) ثم يغلق الملف.

إظهار بيانات اعتماد

الاختيار الرابع من قائمة الاعتمادات هو الاستفسار عن بيانات اعتماد. ويتم الاستفسار عن بيانات الاعتماد بإدخال رقمه. فإذا كان الرقم غير موجود ظهرت رسالة خطأ تفيد عدم وجود هذا الرقم وتطلب إدخال الرقم الصحيح أما إذا كان الرقم موجوداً بالملف فإن بيانات الاعتماد تظهر على الشاشة المعدة لذلك. وتظهر البيانات بدون صلاحيات التعديل مثلما تظهر في برنامج الحذف. والبرنامج الذي يتولى مهمة إظهار بيانات اعتماد هو ORDDIS.PRG ويشتمل شكل ١٨ - ٤ على



شكل ١٨ - ٤ خريطة برنامج ORDDIS.PRG

الخريطة التي توضح علاقة هذا البرنامج بالبرامج الأخرى داخل النظام أما شكل ١٩ - ٤ فيشتمل على محتويات هذا البرنامج.

```

1 * -----*
2 * Program   : ORDDIS.PRG                      *
3 * Author    : Magdi M. Abu Al-Ata            *
4 * Date      : October 24, 1990                 *
5 * Purpose   : To display an existing order    *
6 * Called from: ORDMENU.PRG                     *
7 * Called from: DBAPP.PRG                       *
8 * -----*
9 USE order INDEX order    && Indexed on ord_no
10 DO WHILE .T.
11     DO border WITH "Order main data", "Displaying order data"
12     DO ord_fmt
13     @24,0
14     @ 24,24 SAY "Press Enter to return to the menu"
15     @ 10,28 GET m_ord_no
16     READ
17     @24,0
18     SEEK m_ord_no
    
```

شكل ١٩ - ٤ برنامج ORDDIS.PRG

```

19      DO CASE
20      CASE m_ord_no = SPACE(5)
21      <-----EXIT
22      CASE .NOT. FOUND()
23      | DO error WITH "Order No. " + TRIM(m_ord_no) + " Not fond."
24      -----LOOP
25      CASE FOUND()
26      @ 24,0
27      @ 10, 64 GET ORD_VAL PICTURE "9,999.99"
28      @ 11, 28 GET ORD_KIND
29      @ 11, 64 GET ORD_DAT
30      @ 12, 28 GET ORD_DESC
31      @ 12, 64 GET SHIP_NAM
32      @ 13, 28 GET COMP_NAM
33      @ 13, 64 GET COMP_ADR
34      @ 14, 28 GET BNK_NAM
35      @ 14, 64 GET BNK_ADR
36      @ 15, 28 GET BILL_NO
37      @ 15, 64 GET DOC_NO
38      @ 18, 17 GET COMISION PICTURE "9,999.99"
39      @ 18, 38 GET GOOD_INS PICTURE "9,999.99"
40      @ 18, 65 GET POSTAGE PICTURE "9,999.99"
41      @ 19, 38 GET LBR_TRNS PICTURE "9,999.99"
42      @ 19, 65 GET CUSTOM PICTURE "9,999.99"
43      @ 9, 2 TO 16, 77
44      @ 17, 2 TO 21, 77
45      @ 20, 17 SAY total PICTURE "9,999.99"
46      CLEAR GETS
47      ENDCASE
48      * Determine if the operator wants to display another.
49      STORE " " TO m_ok
50      @ 24, 0
51      @ 24, 20 SAY "Display another order? (Y/N)-----> " GET m_ok
52      READ
53      IF UPPER (m_ok) = "N"
54      <-----EXIT
55      ENDIF
56      @24,0
57 ENDDO
58 CLOSE DATABASES
59 RETURN
60
61 *****
62 * End of file ORDNUM.PRG *
63 *****

```

تابع شكل ١٩ - ٤

يستخدم هذا البرنامج الشاشة المعدة لإدخال أو تعديل البيانات (راجع شكل ٦ - ٤). وهذا البرنامج شبيه ببرنامج التعديل السابق والاختلاف بينهما أننا هنا نستخدم أمر CLEAR GETS بعد إحضار حقول السجل لإلغاء صلاحيات تعديلها

(سطر رقم ٤٦) أما في برنامج ORDED.T.PRG فإننا نستخدم أمر READ لإيقاف تنفيذ البرنامج ووضع المؤشر عند أول حقل في مجموعة أوامر GET...@ لتعديلها. وكذلك يتم تعديل حقل إجمالي المصروفات (TOTAL) بعد عملية التعديل لأن التعديل ربما طرأ على أحد بنود المصروفات أما في برنامج الإظهار فلا داعي لتعديل هذا الحقل لأن الأمر اقتصر على الإظهار فقط.

تقارير الاعتمادات

تعتبر التقارير أحد الاختيارات الهامة في أي نظام لإدارة قواعد البيانات. وعادة يلزم لأي نظام أكثر من تقرير. والتقارير الواردة في هذا النظام بسيطة وتشتمل أيضاً على أفكار بسيطة وذلك لأننا نأخذ في الكتاب بمبدأ التدرج في الأنظمة التي يشتمل عليها من الأسهل إلى الأصعب وباعتبار أن هذا هو أول نظام نتناوله فقد عمدنا إلى التبسيط إلا أننا سنتعرض في الفصول القادمة إن شاء الله لتقارير أكثر تعقيداً تشتمل على أفكار كثيرة لاستخراج التقارير على الشاشة أو طباعتها على الطابعة. ومطلوب في النظام الذي بين أيدينا ثلاثة تقارير ولذلك لجأنا إلى إعداد قائمة مستقلة داخل النظام للتقارير. والاختيار الذي يتيح الحصول على هذه القائمة من قائمة الاعتمادات هو الاختيار الخامس (Reports) ويوضح شكل ٢٠ - ٤ قائمة التقارير المطلوبة وهي عبارة عن التقارير الآتية:

- ١ - تقرير بالاعتمادات خلال فترة معينة
- ٢ - تقرير بالاعتمادات قبل تاريخ معين
- ٣ - تقرير بالاعتمادات بعد تاريخ معين

ولأن بيانات التقارير الثلاثة متشابهة والاختلاف الوحيد بينها هو في تاريخ الاعتماد فقد أعدنا برنامجاً واحداً لهذا الغرض هو برنامج ORDREP.PRG وهو يستدعي إجراءات سبق شرحها بالإضافة إلى إجراء جديد هو ordprd سيتم شرحه بعد الانتهاء من شرح هذا البرنامج.

Tuesday 26, February 1991	Time: 15:82:42
Arabian Computer Center C.R. 39387	
Purchase orders receipt menu	
1. During a period.. 2. Before date..... 3. After date..... 0. Go to orders menu	
Select number =====> █	

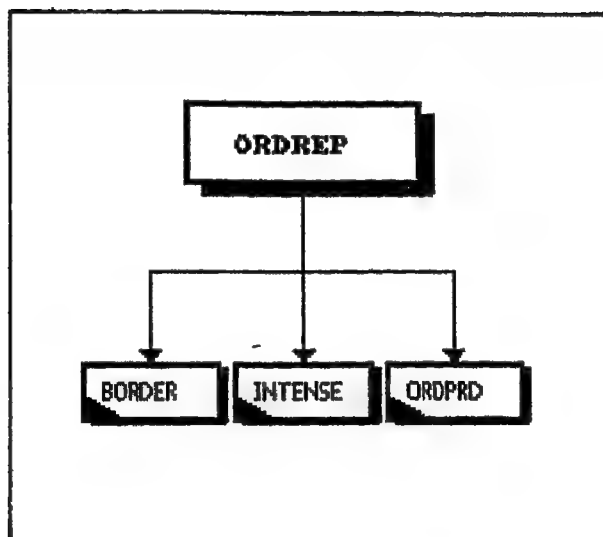
شكل ٢٠ - قائمة تقارير الاعتمادات

ويشتمل شكل ٢١ - ٤ على الخريطة التي توضح علاقة هذا البرنامج بغيره من البرامج داخل النظام. كما يشتمل شكل ٢٢ - ٤ على محتويات هذا البرنامج. وعن هذا البرنامج نوضح ما يلي:

- يستخدم هذا البرنامج ملف ORDER.BDF ومعه ملف الفهرس IDATE.NDX وهو مرتب حسب تواريخ الاعتمادات. (سطر ١١) لأن الاعتمادات تظهر بترتيب تواريخها.

- يتم إنشاء حقول بالذاكرة لقبول التواريخ التي سيتم التعامل معها داخل البرنامج. وهذه الحقول أربعة SDATE لتاريخ بداية فترة التقرير وEDATE لتاريخ نهاية فترة التقرير ويستخدمان إذا طلب التقرير الأول وBDATE وتستخدم إذا طلب التقرير الثاني (بعد تاريخ معين) وADATE وتستخدم إذا طلب التقرير الثالث (قبل تاريخ معين). وتنشأ هذه الحقول الأربعة حرفية.

ينشئ البرنامج دوائر لرسم ش - الاختيارات وقبول الرقم أو الحرف الدال على الاختيار ويتم رسم الشاشة بنفس الطريقة التي شرحناها من قبل (سطر ١٧ - ٢٩).



شكل ٢١ - ٤ خريطة برنامج ORDREP.PRG

```

1 * -----*
2 * Program.....: ORDREP.PRG*
3 * Called form.: ORDMENU.PRG*
4 * Called form.: DBAPP.PRG*
5 * Programmer...: Magdi M. Abu Al-Ata*
6 * Date.....: Dec. 25,1990*
7 * Purpose.....: To display purchase orders according to selected date*
8 * -----*
9 PRIVATE scrtitle,scrfootr,sdate,edate,bdate,adate
10 STORE " " TO yn,mor
11 USE order INDEX idate && Indexed on ord_dat
12 STORE SPACE (8) TO sdate,edate,bdate,adate
13 scrtitle = "Purchase orders reoprt menu"
14 scrfootr = "Select number =====>"
15 ok = .T.
16 DO WHILE ok
17     choice = " "
18     DO border WITH scrtitle,scrfootr
19     * Draw menu
20     @10, 5 SAY "1. During a period.."
21     DO intense WITH 10, 8,"D"
22     @11, 5 SAY "2. Before date....."
23     DO intense WITH 11, 8,"B"
24     @12, 5 SAY "3. After date....."
    
```

شكل ٢٢ - ٤ برنامج ORDREP.PRG

```

25 DO intense WITH 12, 8,"A"
26 @13, 5 SAY "0. Go to orders menu"
27 DO intense WITH 13, 8,"G"
28 @22,43 GET choice
29 READ
30 scrfootr="Orders data"
31 * Do selected option
32 DO CASE
33 CASE choice $ "1Dd"
34     @24,0
35     @24,5 SAY "Enter start date " GET sdate PICTURE "99/99/99"
36     READ
37     @24,40 SAY "Enter end date " GET edate PICTURE "99/99/99"
38     READ
39     sdate = CTOD(sdate)
40     edate = CTOD(edate)
41     dated= "ord_dat >= sdate .AND. ord_dat <= edate"
42     scrtitle="Report period: "+dtoc(sdate)+" to: "+dtoc(edate)
43 CASE choice $ "2bB"
44     @24,0
45     @24,5 SAY "Orders before date: " GET bdate PICTURE "99/99/99"
46     READ
47     bdate = CTOD(bdate)
48     dated="ord_dat <= bdate"
49     scrtitle="Orders before date: "+dtoc(bdate)
50 CASE choice $ "3Aa"
51     @24,0
52     @24,5 SAY "Orders after date: " GET adate PICTURE "99/99/99"
53     READ
54     adate = CTOD(adate)
55     dated="ord_dat >= adate"
56     scrtitle="Orders after date: "+dtoc(adate)
57 CASE choice $ "0gG"
58 <-----RETURN
59 OTHERWISE
60 <-----LOOP
61 ENDCASE
62 SET FILTER TO &dated
63 GO TOP
64 DO ordprd
65 SET FILTER TO
66 scrtitle = "Purchase orders reoprt menu"
67 scrfootr = "Select number =====>"
68 ENDDO
69 RELEASE sdate,edate,bdate,adate,yn,ok,mor
70 CLOSE DATABASES
71 RETURN
72 *****
73 * End of file ORDREP.PRG *
74 *****

```

- يتم تنفيذ واحدة من ثلاث حالات (سطر ٣٢) وبالرغم من أننا يمكننا وضع برنامج مستقل لكل حالة يتم تنفيذه عند اختيار الرقم أو الحرف الدال على الاختيار كما كنا نفعل في برامج القوائم السابقة إلا أننا هنا أدخلنا الأوامر المطلوب تنفيذها مباشرة بعد أمر CASE لقلتها ولتوفير عدد البرامج داخل النظام.
- الحالة الأولى: تقرير خلال فترة زمنية (سطر ٣٣ - ٤٢) ويتم تنفيذها إذا اختار المستفيد الرقم ١ أو حرف D أو حرف d. وفيها يتم الآتي:
 - * ظهور رسالة تطلب إدخال تاريخ بداية الفترة ويتم قبول التاريخ (سطر ٣٥ - ٣٦). ويقبل التاريخ كعبارة حرفية.
 - * ظهور رسالة تطلب إدخال تاريخ نهاية الفترة ويتم قبول التاريخ (سطر ٣٧ - ٣٨).
- * يتم تحويل العبارة الحرفية التي تحتوي على التاريخ إلى عبارة تاريخية وذلك باستخدام الوظيفة () CTOD (سطر ٣٩ - ٤٠).
- * يتم إنشاء حقل ذاكرة يشتمل على الشرط المطلوب وهو أن يكون تاريخ الاعتماد أكبر من أو يساوي التاريخ الذي أدخل لبداية الفترة (SDATE) وفي نفس الوقت أقل من أو يساوي التاريخ الذي أدخل لنهاية الفترة (EDATE) (سطر ٤١).
- * يتم تعديل اسم الشاشة لتدل على فترة التقرير ولذلك تظهر فيها التواريخ التي أدخلت لبداية ونهاية الفترة (سطر ٤٢).
- الحالة الثانية: تقرير قبل تاريخ معين (سطر ٤٣ - ٤٩) ويتم تنفيذها إذا اختار المستفيد الرقم ٢ أو حرف B أو حرف b وتتم فيها إجراءات مشابهة للحالة الأولى مثل قبول التاريخ (سطر ٤٥ - ٤٦) وتحويله إلى عبارة تاريخية (سطر ٤٧) ووضع الشرط المطلوب وهو الاعتمادات التي يساوي تاريخها أو يقل عن تاريخ معين (سطر ٤٨) داخل حقل الذاكرة. وتغيير اسم الشاشة (سطر ٤٩).
- الحالة الثالثة: تقرير بعد تاريخ معين (سطر ٥٠ - ٥٦) ويتم تنفيذها إذا اختار المستفيد الرقم ٣ أو حرف A أو حرف a وهي تشبه الحالة الثانية إلا أن الشرط المحدد هنا عكس الموجود في الحالة الثانية لأننا نريد الاعتمادات التي يساوي

تاريخها أو يزيد عن التاريخ الذي تم إدخاله.
- الحالة الرابعة: ويتم تنفيذها إذا اختار المستفيد الرجوع إلى القائمة السابقة (سطر ٥٧ - ٥٨).

- وخلاف هذه الحالات الأربع ينتقل البرنامج إلى بداية الدوارة (سطر ٥٩ - ٦٠).
- حددنا الشرط الذي سيتم إظهار التقدير طبقاً له ووضعناه في حقل ذاكرة اسمه .
dated

- أمر SET FILTER TO (سطر ٦٢) من الأوامر القوية جداً التي تجعل ملف قاعدة البيانات يبدو كما لو كان يشتمل فقط على السجلات التي تقابل شرط / شروط محددة. وقد استخدم في هذا البرنامج ليستخرج فقط السجلات التي تقابل فترة معينة أو التي تقع قبل أو بعد تاريخ معين. وقد استخدمنا مع الأمر الوظيفة & وتسمى «ماكرو» ومعنى dated & أي محتويات حقل الذاكرة dated. ولأن محتويات dated تختلف حسب الحالة المختارة فكأننا نطلب أن يتم استخراج السجلات التي تقابل الحالة المختارة فقط وكان يمكننا مثلاً استخدام الأمر بالصورة التالية:

SET FILTER TO ORD_DAT <= DATE

لاستخراج السجلات التي تقع قبل تاريخ معين. وكنا في هذه الحالة سنستخدم الأمر عقب كل حالة لوضع الشرط المطلوب. إلا أننا فضلنا استخدام الوظيفة & لأنها أعم في الاستخدام ولتوضيح طريقة استخدامها.

- أمر GO TOP (سطر ٦٣) بعد أمر SET FILTER ليضع المؤشر في أول الملف الذي يشتمل على السجلات المختارة ونصح بوضعه دائماً بعد أمر SET FILTER. في بداية الملف.

- أمر DO ordprd (سطر ٦٤) هو الذي يتولى طباعة صفحات التقرير وستتناوله بالشرح بعد قليل.

- أمر SET FILTER TO (سطر ٦٥) لإعادة الملف إلى حالته الطبيعية أي ليشتمل على كل السجلات الموجودة بعد طباعة التقرير من السجلات المختارة فقط.

- بعد الانتهاء من التقرير تعود عناوين الشاشة لما كانت عليه (سطر ٦٦ - ٦٧).
- قبل انتهاء البرنامج يتم حذف حقول الذاكرة التي أنشئت (سطر ٦٩) وإغلاق الملفات (سطر ٧٠).

الإجراء ordprd

- يشتمل شكل ٢٣ - ٤ على محتويات الإجراء ordprd ومنه تلاحظ ما يلي:
- ينشئ عدادًا للصفحات بقيمة ابتدائية هي 1 (سطر ١٧١).

```

166 *****
167 * Procedure : ordprd
168 * Purpose : Views order(s) according to specific date(s)
169 *****

170 PROCEDURE ordprd
171 page = 1
172 DO WHILE .NOT. EOF()
173     DO border with scrtitle,scrfootr
174     DO ord_fmt
175     @ 6,65 SAY "Page no."+STR(page,3)
176     @ 10, 28 GET ORD_NO
177     @ 10, 64 GET ORD_VAL PICTURE "9,999.99"
178     @ 11, 28 GET ORD_KIND
179     @ 11, 64 GET ORD_DAT
180     @ 12, 28 GET ORD_DESC
181     @ 12, 64 GET SHIP_NAM
182     @ 13, 28 GET COMP_NAM
183     @ 13, 64 GET COMP_ADR
184     @ 14, 28 GET BNK_NAM
185     @ 14, 64 GET BNK_ADR
186     @ 15, 28 GET BILL_NO
187     @ 15, 64 GET DOC_NO
188     @ 18, 17 GET COMISION PICTURE "9,999.99"
189     @ 18, 38 GET GOOD_INS PICTURE "9,999.99"
190     @ 18, 65 GET POSTAGE PICTURE "9,999.99"
191     @ 19, 38 GET LBR_TRNS PICTURE "9,999.99"
192     @ 19, 65 GET CUSTOM PICTURE "9,999.99"
193     @ 20, 17 SAY TOTAL PICTURE "9,999.99"
194     CLEAR GETS
195     @ 9, 2 TO 16, 77
196     @ 17, 2 TO 21, 77
197     @ 24,2 SAY " Display more records? [Y/N] " GET mor PICTURE "I"
198     READ
199     IF mor = "N"
200     <-----EXIT

```

شكل ٢٣ - ٤ الإجراء ordprd

```

201      ENDIF
202      SKIP
203      page=page+1
204      IF EOF()
205          @24,2
206          @24,2 SAY "***** End of data. Press any key *****"
207          READ
208      ENDIF
209 ENDDO
210 RETURN
211

```

تابع شكل ٢٣ - ٤

- ينشئ دارة لطباعة محتويات الاعتماد تتكرر هذه الدارة حتى ينتهي الملف أو يجيب المستخدم على رسالة Display more records? (Y/N) باختيار N (سطر ١٧٢ - ٢٠٩).
- بعد إظهار محتويات الاعتماد ينتقل المؤشر إلى السجل التالي دائماً (سطر ٢٠٢) ويزاد عدد الصفحات مقدار واحد (سطر ٢٠٣).
- إذا انتهت سجلات الملف أثناء العرض تظهر رسالة للمستخدم تفيد ذلك حتى يفهم أن البيانات انتهت (سطر ٢٠٤ - ٢٠٨).

الفصل الخامس

المخازن

Inventory

يشرح هذا الفصل وظائف ومراقبة المخزون الرئيسي مثل إضافة أو تعديل أو حذف أحد الأصناف الموجودة بالمخازن بالإضافة إلى الاستفسارات والتفاريير عن صنف معين أو أصناف المخازن أو الأصناف التي بلغت نقطة إعادة الطلب.

والنظام وهذه لا يحقق الرقابة المطلوبة على المخازن وإنما يلزم ربطه مع نظام المشتريات والمبيعات اللذان سنشرحهما في الفصلين القادمين.

ويبدأ الفصل بشرح وظائف النظام ثم يشرح الملفات المطلوبة ثم يشرح بالتفصيل كل برنامج من برامج النظام على حدة وعلاقته بغيره من البرامج الأخرى داخل النظام.

تحتاج شركة الحاسب العربي لنظام لمراقبة المخزون من الحاسبات وقطع غيارها. ويرتبط أي نظام للمخازن أو المستودعات بنظام المشتريات والمبيعات لأن أرصدة المخازن يجب أن تتعدل دائماً بعد كل عملية شراء لتضاف الأصناف المشتراة إلى رصيدها الموجود بالمخازن، وكذلك بعد كل عملية بيع لتطرح الكميات المباعة من أرصدة المخازن لكي تعكس الأرصدة الموجودة بالمخازن القيمة الحقيقية للأصناف. ولذلك فإن فهم نظام الرقابة على المخازن أو المستودعات فهماً جيداً يستلزم منك فهم الفصلين التاليين وهما نظام المشتريات ونظام المبيعات.

والنظام الذي سنشرحه في هذا الفصل خاص بالتعامل مع المخزون الرئيسي ويقوم بالوظائف التالية:

- ١ - إضافة صنف جديد إلى الأصناف الموجودة بالمخازن.
- ٢ - تعديل بيانات صنف موجود.
- ٣ - حذف صنف من ملف المخزون الرئيسي لانتهاء العمل به.
- ٤ - الاستفسارات والتقارير وتشمل الاستفسار عن أي صنف موجود بالمخازن أو تقرير بالموجود في المخازن في أي لحظة أو بالأصناف التي وصلت نقطة إعادة الطلب.

أما البرنامج اللازم لتعديل أرصدة المخازن بإضافة الأصناف المشتراة إلى المخزون الرئيسي بعد كل عملية شراء أو طرح الأصناف المباعة من المخزون الرئيسي بعد كل عملية بيع فسنرجئه إلى الفصل التاسع ضمن برامج صيانة الملفات.

وفيما يلي نوضح الخطوات والبرامج المطلوبة لتصميم نظام يحقق الوظائف السابقة.

تصميم ملف قاعدة البيانات

كما تلاحظ من الوظائف السابقة أنها تحتاج للتعامل مع ملف واحد. وهو ملف المخزون الرئيسي. ولذلك فسيقصر الشرح في هذا الفصل أيضاً على البرامج التي

تتعامل مع ملف واحد. وابتداءً من الفصل القادم بإذن الله سنعرف كيف يتم التعامل مع أكثر من ملف.

والملف المطلوب لهذا النظام هو ملف INV. DBF لإنشاء هذا الملف أدخل الأمر الآتي من نقطة المحث

. CREATE INV

ثم أدخل المواصفات الموجودة بشكل ١ - ٥. ومنه يتضح أن ملف المخزون الرئيسي يجب أن يشتمل على البيانات الأساسية للأصناف الموجودة بالمخازن وتشمل:

Bytes remaining: 3943

CURSOR ← → Char: ← → Word: Home End Pan: ^← ^→	INSERT Char: Ins Field: ^N Help: F1	DELETE Char: Del Word: ^V Field: ^U	Up a field: ↑ Down a field: ↓ Exit/Save: ^End Abort: Esc
--	---	---	---

Field Name	Type	Width	Dec	
1 ITEM_NO	Character	5		
2 DESC	Character	12		
3 MODEL	Character	10		
4 LOCAT	Character	5		
5 ONSTOCK	Numeric	5	0	
6 PRICE	Numeric	7	2	
7 UPDT_DAT	Date	8		
8 REORDER	Numeric	4	0	

MODIFY STRUCTURE <C> INV Field: 0/0

Enter the field name.

Field names begin with a letter and may contain letters, digits and underscores

شكل ١ - ٥ مواصفات ملف المخازن

رقم الصنف (ITEM_NO) - اسم الصنف (DESC) - الموديل (MODEL) - مكان الصنف داخل المخزن (LOCAT) - الكمية الموجودة بالمخازن (CONSTOCK) - السعر (PRICE) - تاريخ آخر تعديل (UPDT_DAT) - نقطة إعادة الطلب . (REORDER)

إنشاء ملفات الفهرسة

سنحتاج لترتيب ملف المخزون الرئيسي بطريقتين:
 الأولى: طبقاً لرقم الصنف ويستخدم هذا الرقم لتمييز الأصناف عن بعضها في حالة إضافة صنف جديد للمخازن أو الاستعلام عنه أو تعديل بياناته.
 الثانية: طبقاً لاسم الصنف وذلك لتسهيل استخراج التقارير مرتبة حسب أيجديات أسماء الأصناف.

ويوضح شكل ٢ - ٥ أسماء ملفات الفهرسة والحقول التابعة لها المستخدمة في النظام.

ملف قاعدة البيانات Database file	حقل الفهرسة Key	اسم ملف الفهرس Index file	طريقة ترتيب الملف Indexed By
INV	ITEM_No DESC	ITEM.NDX DESC.NDX	طبقاً لرقم الصنف طبقاً لاسم الصنف

شكل ٢ - ٥ أسماء ملفات الفهرسة والحقول التابعة لها

أنشئ ملفات الفهرسة بعد الانتهاء من إدخال مواصفات ملف قاعدة البيانات وحفظها بالأمرين التاليين:

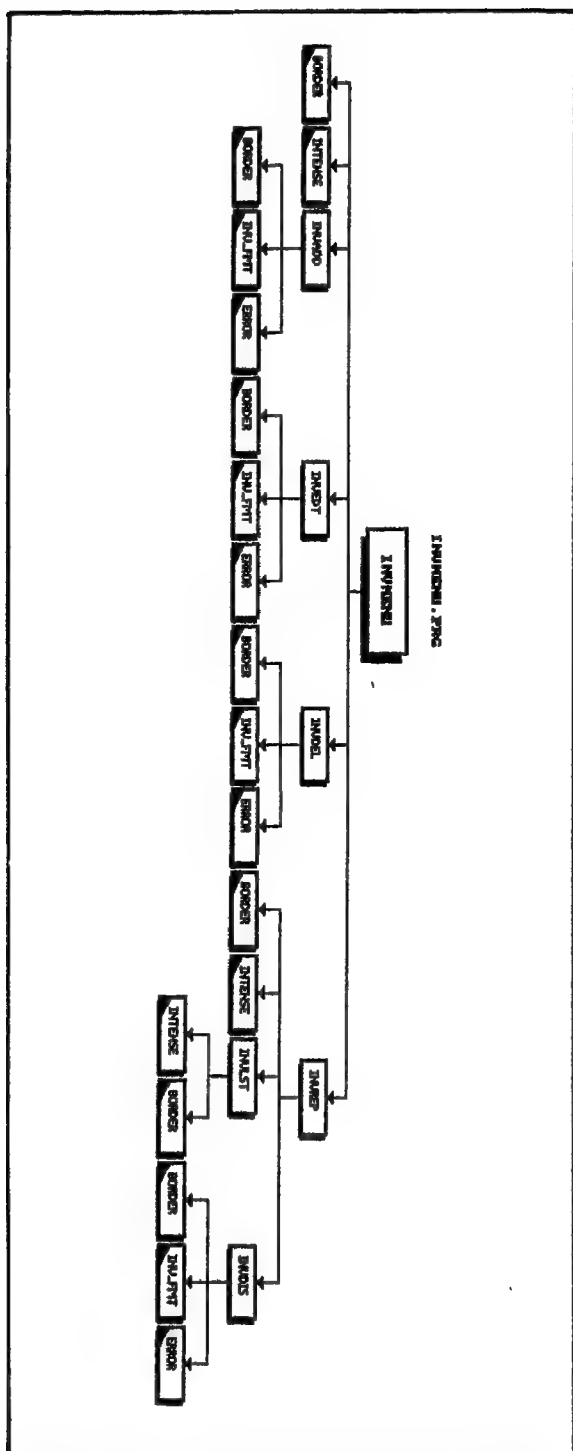
. INDEX ON ITEM_NO TO ITEM

. INDEX ON DESC TO DESC

وسبق أن أوضحنا أن إنشاء ملف الفهرس لا يشترط وجود بيانات بملف قاعدة البيانات.

تحديد برامج النظام

بعد أن حددنا الهدف من النظام واختارنا الملفات المطلوبة يجب أن نعد خريطة للنظام توضح البرامج التي يشتمل عليها بحيث تشتمل على الوظائف المطلوبة من النظام. ويوضح شكل ٣ - ٥ خريطة النظام التي توضح العلاقة بين البرامج التي



شكل ٣ - ٥ خريطة برنامج INVENTU.PRG

يشتمل عليها. وفي هذه الخريطة تم تقسيم النظام إلى وظائف أساسية هي الإضافة (Adding) والتعديل (Editing) والحذف (Deleting) والتقارير (Reports). ويمثل كل وظيفة من هذه الوظائف مستطيل داخل خريطة النظام. وستتناول فيما يلي شرح البرامج المطلوبة لهذه الوظائف بالإضافة إلى إجراء جديد ضمن ملف الإجراءات.

قائمة نظام المخزون الرئيسي

يجب أن تلمي قائمة المخزون الرئيسي الوظائف الأساسية الموضحة بخريطة النظام السابقة. ويوضح شكل ٤ - ٥ القائمة الرئيسية لنظام المخازن والاختيارات التي تشتمل عليها. كما يشتمل شكل ٥ - ٥ على برامج INVMENU.PRG اللازم لإظهار هذه القائمة وتنفيذ اختياراتها ولا يختلف هذا البرنامج عن برنامج قائمة الاعتمادات المستندية الذي تقدم في الفصل السابق.

Tuesday 26, February 1991	Time: 15:05:23
Arabian Computer Center C.R. 39987	
<div style="border: 1px solid black; padding: 5px; margin: 5px 0; text-align: center;">Inventory menu</div> <div style="border: 1px solid black; padding: 5px;"> <p>(1) Add new item (a).....</p> <p>(2) Edit existind item (e)</p> <p>(3) Delete item (s).....</p> <p>(4) Reports.....</p> <p>(G) Go to main menu.....</p> <p>(Q) Quit to DOS.....</p> </div> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> Select number or colored letter ==> █ </div>	

شكل ٤ - ٥ القائمة الرئيسية لنظام المخازن

```

1 * -----*
2 * Program.....: invmenu.PRG*
3 * Called form.: dbapp.prg*
4 * Programmer...: Magdi M. Abu Al-Ata*
5 * Date.....: Dec. 25,1990*
6 * Purpose.....: To display inventory menu*
7 * -----*
8 PRIVATE scrtitle,scrfootr,choice
9 scrtitle = "Inventory menu"
10 scrfootr = "Select number or colored letter ==>"
11 ok = .T.
12 DO WHILE ok
13     choice = " "
14     DO border WITH scrtitle,scrfootr
15     * Draw menu
16     @10, 5 SAY "(1) Add new item (s)....."
17     DO intense WITH 10,10,"A"
18     @11, 5 SAY "(2) Edit existind item (s)"
19     DO intense WITH 11,10,"E"
20     @12, 5 SAY "(3) Delete item (s)....."
21     DO intense WITH 12,10,"D"
22     @13, 5 SAY "(4) Reports....."
23     DO intense WITH 13,10,"R"
24     @14, 5 SAY "(G) Go to main menu....."
25     DO intense WITH 14,10,"G"
26     @15, 5 SAY "(Q) Quit to DOS....."
27     DO intense WITH 15,10,"Q"
28     @22,43 GET choice
29     READ
30     * Do selected option
31     DO CASE
32     CASE choice $ "1aA"                && If choice = 1 or a or A
33         DO invadd                      && Add new item
34     CASE choice $ "2eE"                && If choice = 2 or e or E
35         DO invedt                      && Edit an item
36     CASE choice $ "3dD"                && If choice = 3 or d or D
37         DO invdel                      && Delete an item
38     CASE choice $ "4rR"                && If choice = 4 or r or R
39         DO invrep                      && Display reports menu
40     CASE UPPER(choice) = "G"
41     <-----RETURN                      && Go to Main menu
42     CASE UPPER(choice) = "Q"
43     CLEAR
44     <=====QUIT                        && Quit to DOS
45     OTHERWISE
46     <-----LOOP
47     ENDCASE
48 ENDD
49 *****
50 * End of file INVMENU.PRG *
51 *****

```

شكل ٥ - ٥ برنامج INVMENU.PRG

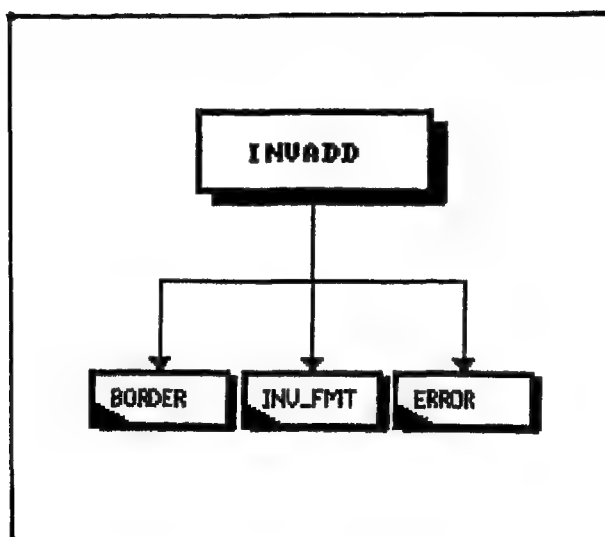
إضافة صنف جديد إلى أصناف المخازن

الاختيار الأول في قائمة نظام المخازن الرئيسي هو إضافة صنف جديد والبرنامج اللازم لإضافة صنف جديد هو INVADD.PRG. ويشتمل شكل ٦ - ٥

Tuesday 26, February 1991		Time: 15:11:32	
Arabian Computer Center C.R. 39387			
Inventory's Item main data			
Item No.	<input type="text"/>		
Description	<input type="text"/>		
Model	<input type="text"/>		
Location	<input type="text"/>		
Quantity on stock	<input type="text"/>		
Unit price	<input type="text"/>	Reorder point	<input type="text"/>
Adding a new item			
Press Enter to return to the menu			

شكل ٦ - ٥ شاشة إدخال بيانات الأصناف

على شاشة إدخال بيانات الأصناف كما يشتمل شكل ٧ - ٥ على الخريطة التي توضح علاقة هذا البرنامج بالبرامج الأخرى داخل النظام. ويشتمل شكل ٨ - ٥ على محتويات هذا البرنامج. ولا تختلف فكرة هذا البرنامج عن برنامج الإضافة الذي شرحناه في الفصل السابق. ويستدعى البرنامج الإجراء inv_fmt لرسم شاشة إدخال البيانات ويشتمل شكل ٩ - ٥ على الإجراء inv_fmt وقد شرحنا سابقاً أنه بإمكانك كتابة البرنامج أمراً أو استخدام أمر CREATE SCREEN INV_FMT لتصميم الشاشة ثم تعديل ملف INV_FMT.FMT (راجع الفصل السابق).



شكل ٧ - ٥ خريطة برنامج INVADD.PRG

```

1 * -----*
2 * Program   : INVADD.PRG                      *
3 * Author    : Magdi M. Abu Al-Ata             *
4 * Date      : October 24, 1990                 *
5 * Purpose   : Adds new item                   *
6 * Called from: INVMENU.PRG                     *
7 * Called from: DPAPP.PRG                       *
8 * -----*
9 USE inv INDEX item      && Indexed on item_no
10 DO WHILE .T.
11     DO border WITH "Inventory's Item main data", "Adding a new item"
12     DO inv_fmt
13     @24,0
14     @ 24,23 SAY "Press Enter to return to the menu"
15     @ 10,28 GET m_item
16     READ
17     SEEK m_item
18     DO CASE
19     CASE m_item = SPACE(5)
20     <-----EXIT
21     CASE FOUND()
22     | DO error WITH "Item No. " + TRIM(m_item) + " already exist."
23     <-----LOOP
24     CASE .NOT. FOUND()
25     | APPEND BLANK
    
```

شكل ٨ - ٥ برنامج INVADD.PRG

```

26      @ 24,0
27      REPLACE item_no with m_item
28      ENDCASE
29      @ 24,0
30      @ 12, 28 GET m_desc
31      @ 14, 28 GET m_model
32      @ 16, 28 GET m_locat
33      @ 18, 28 GET m_on      PICTURE "99999"
34      @ 20, 28 GET m_price  PICTURE "9,999.99"
35      @ 20, 60 GET m_reorder PICTURE "99999"
36      READ
37      STORE " " TO m_ok
38      @ 24, 24 SAY "Valid item data? (Y/N)---) "GET m_ok
39      READ
40      IF UPPER (m_ok) = "Y"
41          @ 24,0
42          REPLACE item_no with m_item, desc with m_desc, ;
42          model with m_model, locat with m_locat, ;
42          onstock with m_on, price with m_price, ;
42          reorder with m_reorder
43      ELSE
44          DELETE          && Delete the APPENDED record
45      ENDIF
46 ENDDO
47 CLOSE DATABASES
48
49 *****
50 * End of file INVADD.PRG *
51 *****

```

تابع شكل ٨ - ٥ برنامج INVADD.PRG

```

212 *****
213 * Procedure : inv_fmt
214 * Purpose : Drows item(s) screen format
215 *****
216 PROCEDURE inv_fmt
217 PUBLIC m_item, m_locat, m_desc, m_model, m_on, m_price, m_reorder
218 * Gives those variables their starting values
219 STORE SPACE(5) TO m_item, m_locat
220 STORE SPACE(12) TO m_desc
221 STORE SPACE(10) TO m_model
222 STORE 0 TO m_on, m_price, m_reorder
223 @ 10, 6 SAY "Item No."
224 @ 10, 28 224 @ 10, 28 GET m_item
225 @ 12, 6 SAY "Description"

```

شكل ٩ - ٥ الإجراء inv_fmt

```

226 @ 12, 28 GET m_desc
227 @ 14, 6 SAY "Model"
228 @ 14, 28 GET m_model
229 @ 16, 6 SAY "Location"
230 @ 16, 28 GET m_locat
231 @ 18, 6 SAY "Quantity on stock"
232 @ 18, 28 GET m_on PICTURE "99999"
233 @ 20, 6 SAY "Unit price"
234 @ 20, 28 GET m_price PICTURE "9,999.99"
235 @ 20, 44 SAY "Reorder point"
236 @ 20, 60 GET m_reorder PICTURE "99999"
237 CLEAR GETS
238 RETURN
239

```

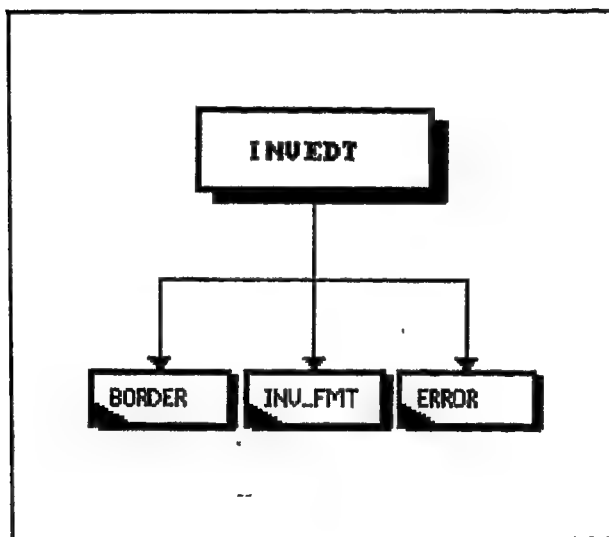
تابع شكل ٩ ..

تعديل بيانات المخازن

الاختيار الثاني في قائمة نظام المخازن الرئيسي هو تعديل بيانات صنف موجود ويستخدم برنامج التعديل نفس الشاشة المستخدمة لغرض الإضافة ولا تختلف فكرته عن برنامج التعديل الموجود في الفصل السابق. ولذلك لا نرى ضرورة لإعادة شرح محتويات البرنامج هنا ويشتمل شكل ١٠ - ٥ على الخريطة التي توضح علاقة البرنامج بغيره من البرامج داخل النظام كما يشتمل شكل ١١ - ٥ على محتويات برنامج INVEDT.PRC اللازم لهذا الغرض.

تعديل برنامج INVEDT.PRC

الاختيار الثالث في قائمة نظام المخازن هو حذف صنف موجود. والبرنامج الذي يقوم بعملية الحذف هو INVDEL.PRG ويشتمل شكل ١٢ - ٥ على الخريطة التي توضح علاقة هذا البرنامج بغيره من البرامج داخل النظام كما يشتمل شكل ١٣ - ٥ على محتويات هذا البرنامج. ويستخدم هذا البرنامج نفس الشاشة المستخدمة لأغراض الإضافة والتعديل. ولا تختلف فكرته عن برنامج الحذف السابق والذي شرحناه في الفصل السابق.



شكل ١٠ - ٥ خريطة برنامج INVEDT.PRG

```

1 * -----*
2 * Program   : INVEDT.PRG                      *
3 * Author    : Magdi M. Abu Al-Ata             *
4 * Date      : October 24, 1990                 *
5 * Purpose   : To edit an existing item        *
6 * Called from: INVMENU.PRG                     *
7 * Called from: DBAPP.PRG                       *
8 * -----*
9 USE inv INDEX item          && Indexed on item_no
10 DO WHILE .T.
11 |   DO border WITH "Inventory's Item main data", "Modifying item data"
12 |   DO inv_fmt
13 |   @24,0
14 |   @ 24,24 SAY "Press Enter to return to the menu"
15 |   @ 10,28 GET m_item
16 |   READ
17 |   @24,0
18 |   SEEK m_item
19 |   DO CASE
20 |   CASE m_item = SPACE(5)
21 |   <-----EXIT
22 |   CASE .NOT. FOUND()
23 |   |   DO error WITH "Item No. " + TRIM(m_item) + " Not fond."
24 |   -----LOOP
25 |   CASE FOUND()
    
```

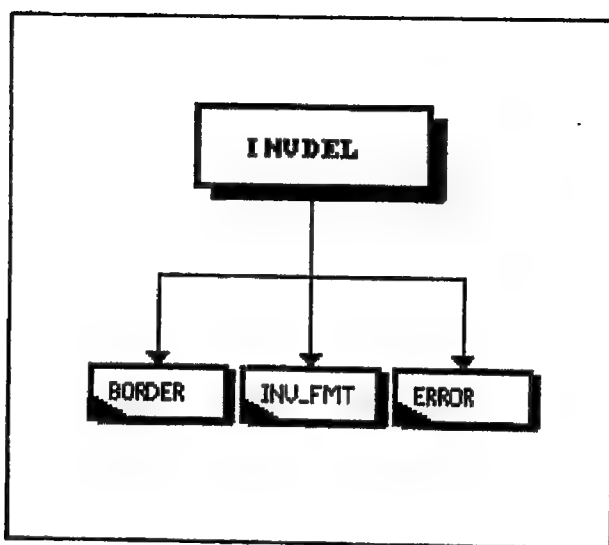
شكل ١١ - ٥ برنامج INVEDT.PRG

```

26      @ 10, 28 GET item_no
27      @ 12, 28 GET desc
28      @ 14, 28 GET model
29      @ 16, 28 GET locat
30      @ 18, 28 GET onstock PICTURE "99999"
31      @ 20, 28 GET price PICTURE "9,999.99"
32      @ 20, 60 GET reorder PICTURE "99999"
33      READ
34      ENDCASE
35      * Determine if the operator wants to edit another.
36      STORE " " TO m_ok
37      @ 24, 0
38      @ 24, 20 SAY "Edit another item record? (Y/N)----> " GET m_ok
39      READ
40      IF UPPER (m_ok) = "N"
41      <-----EXIT
42      ENDIF
43      @24,0
44 ENDDO
45 CLOSE DATABASES
46 RETURN
47
48 *****
49 * End of file INVEDT.PRG *
50 *****

```

تابع شكل ١١ - ٥ برنامج INVEDT.PRG



شكل ١٢ - ٥ خريطة برنامج INUDEL.PRG

```

1 * -----*
2 * Program      : INVDEL.PRG                      *
3 * Author       : Magdi M. Abu Al-Ata            *
4 * Date        : October 24, 1990                 *
5 * Purpose      : To delete an existing item      *
6 * Called from: INVMENU.PRG                       *
7 * Called from: DPAPP.PRG                         *
8 * -----*
9 USE inv INDEX item          && Indexed on item_no
10 DO WHILE .T.
11     DO border WITH "Inventory's Item main data", "Deleting item data"
12     DO inv_fmt
13     @24,0
14     @ 24,24 SAY "Press Enter to return to the menu"
15     @ 10,28 GET m_item
16     READ
17     @24,0
18     SEEK m_item
19     DO CASE
20     CASE m_item = SPACE(5)
21     <-----EXIT
22     CASE .NOT. FOUND()
23     | DO error WITH "Item No. " + TRIM(m_item) + " Not fond."
24     <-----LOOP
25     CASE FOUND()
26     | @ 10, 28 GET item_no
27     | @ 12, 28 GET desc
28     | @ 14, 28 GET model
29     | @ 16, 28 GET locat
30     | @ 18, 28 GET onstock PICTURE "99999"
31     | @ 20, 28 GET price PICTURE "9,999.99"
32     | @ 20, 60 GET reorder PICTURE "99999"
33     | CLEAR GETS
34     ENDCASE
35     * Confirm deletion
36     STORE " " TO m_ok
37     @ 24,0
38     @ 24, 21 SAY "Confirm delete this item? (Y/N)-----> " GET m_ok
39     READ
40     IF UPPER (m_ok) = "Y"
41     | DELETE
42     ENDF
43     * Determine if the operator wants to delete another.
44     STORE " " TO yn
45     @ 24, 0
46     @ 24, 21 SAY "Delete another item record? (Y/N)----> " GET yn
47     READ
48     IF UPPER (yn) = "N"
49     <-----EXIT
50     ENDF
51     @24,0
52 ENDDO
53 CLOSE DATABASES

```

```
54 RETURN
55
56 *****
57 * End of file INVDEL.PRG *
58 *****
```

تابع شكل ١٣ - ٥ برنامج INVDEL.PRG

تقارير المخازن

يتسبب الاختيار الرابع في قائمة المخازن في الحصول على قائمة تقارير المخازن ويوضح شكل ١٤ - ٥ قائمة التقارير المطلوبة وهي عبارة عن ثلاثة تقارير هي:

- ١ - تقرير بكل الأصناف الموجودة.
- ٢ - استفسار عن صنف برقمه.
- ٣ - تقرير بالأصناف التي وصلت إلى نقطة إعادة الطلب.

ويمكنك إضافة تقارير أخرى حسب طلبك مثل تقارير عن الأصناف التي

Tuesday 26, February 1991Time: 15:14:04

Arabian Computer Center
C.R. 39387

Inventory report menu

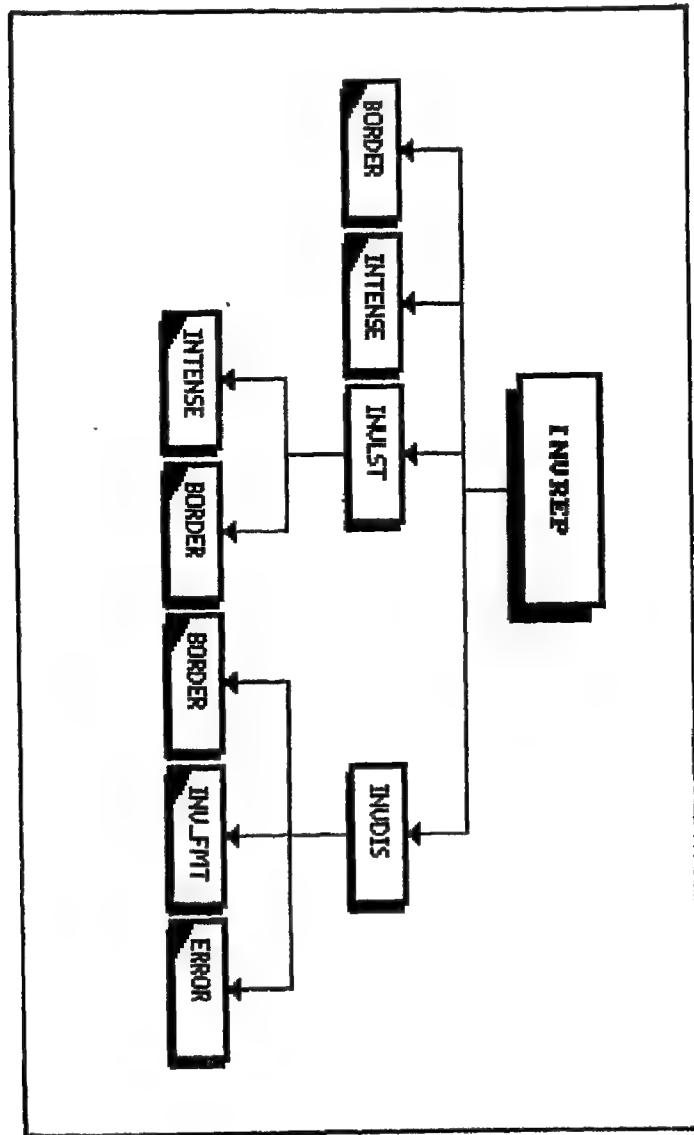
(1) All items.....
(2) Item by number.....
(3) Reorder point.....
(G) Go to inventory menu..
(Q) Quit to DOS.....

Select number or colored letter ==> █

شكل ١٤ - ٥ قائمة تقارير المخازن

تزيد عن سعر معين أو الموجودة بمكان ما داخل المخازن... وهكذا وذلك بالاستعانة بالأفكار الموجودة في البرامج التي سنشرحها هنا.

والبرنامج اللازم لإظهار القائمة الموجودة في شكل ١٤ - ٥ هو INVREP.PRG ويوضح شكل ١٥ - ٥ خريطة البرنامج والبرامج التي يستدعيها بالإضافة إلى



شكل ١٥ - ٥ خريطة برنامج INVREP.PRG

```

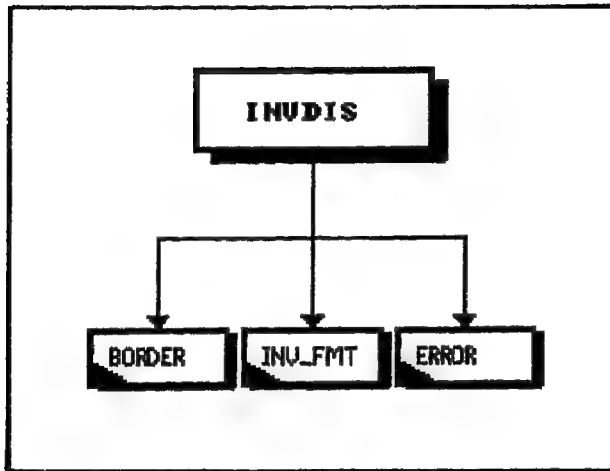
1 * -----*
2 * Program.....: invrep.prg*
3 * Called form.: invmenu.prg*
4 * Called form.: dbapp.prg*
5 * Programmer...: Magdi M. Abu Al-Ata*
6 * Date.....: Dec. 25,1990*
7 * Purpose.....: To display inventory reports menu*
8 * -----*
9 PRIVATE scrtitle,scrfootr
10 scrtitle = "Inventory report menu"
11 scrfootr = "Select number or colored letter ==>"
12 USE Inv INDEX desc    && Indexed on desc
13 ok = .T.
14 DO WHILE ok
15     choice = " "
16     DO border WITH scrtitle,scrfootr
17     * Draw menu
18     @10, 5 SAY "(1) All items....."
19     DO intense WITH 10,10,"A"
20     @11, 5 SAY "(2) Item by number....."
21     DO intense WITH 11,10,"I"
22     @12, 5 SAY "(3) Reorder point....."
23     DO intense WITH 12,10,"R"
24     @13, 5 SAY "(G) Go to inventory menu.."
25     DO intense WITH 13,10,"G"
26     @14, 5 SAY "(Q) Quit to DOS....."
27     DO intense WITH 14,10,"Q"
28     @22,43 GET choice
29     READ
30     * Do selected option
31     DO CASE
32     CASE choice $ "1aA"                && If choice = 1 or a or A
33     | DO invlst                        && List all items
34     CASE choice $ "2iI"                && If choice = 2 or I or i
35     | DO invdis
36     CASE choice $ "3rR"                && If choice = 3 or r or R
37     | SET FILTER TO onstock <= reorder
38     | GO TOP
39     | DO invlst                        && List one item
40     | SET FILTER TO
41     CASE UPPER(choice) = "G"
42     <-----RETURN                    && Go to Inventory menu
43     CASE UPPER(choice) = "Q"
44     5 CLEAR
45     <=====QUIT                    && Quit to DOS
46     OTHERWISE
47     <-----LOOP
48     ENDCASE
49 ENDD
50 *****
51 * End of file INVREP.PRG *
52 *****

```

الإجراءات الأخرى ويشتمل شكل ١٦ - ٥ على محتويات البرنامج . ومنه تلاحظ أن الاختيار الأول والثالث كلاهما يستدعي برنامج INVLIST.PRG . والفرق بينهما أن الاختيار الثالث يختار السجلات التي تشتمل على الأصناف التي بلغ رصيدها نقطة إعادة الطالب أو أقل قبل طباعة التقرير ويستخدم لهذا الغرض أمر SET FILTER الذي شرحناه في الفصل السابق . أما الاختيار الأول فيطبع التقرير بكل الأصناف الموجودة في المخازن . والاختيار الثاني يستدعي برنامجاً مختلفاً وهو INVDIS.PRG لطباعة الصنف باستخدام رقمه .

برنامج طباعة صنف واحد (INVDIS.PRG)

هذا البرنامج يشبه برنامج التعديل السابق ، والفرق بينهما أن هذا البرنامج يلغي صلاحيات التعديل باستخدام أمر CLEAR GETS بعد رسم الشاشة بدلاً من أمر READ . والبرنامج يستخدم الشاشة المستخدمة لأغراض الإضافة والتعديل والحذف . ويشتمل شكل ١٧ - ٥ على خريطة البرنامج وهي تشبه خريطة برنامج INVEDIT.PRG بينما يشتمل شكل ١٨ - ٥ على محتويات البرنامج . ولذلك لن نجد صعوبة في مراجعة منطق البرنامج والإجراءات التي يشتمل عليها .



شكل ١٧ - ٥ خريطة برنامج INVDIS.PRG

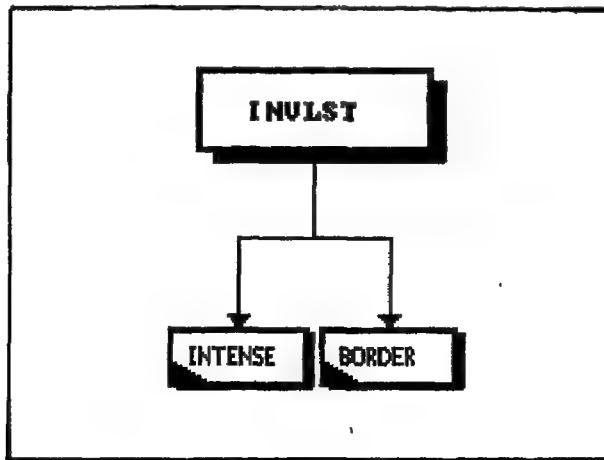
```

1 * -----*
2 * Program   : INVDIS.PRG                      *
3 * Author    : Magdi M. Abu Al-Ata             *
4 * Date      : October 24, 1990                 *
5 * Purpose    : To view an existing item        *
6 * Called from: INVREP.PRG                     *
7 * Called from: INVMENU.PRG                    *
8 * -----*
9 USE inv INDEX item          && Indexed on item_no
10 DO WHILE .T.
11     DO border WITH "Inventory's Item main data", "Displaying item data"
12     DO inv_fmt
13     @24,0
14     @ 24,24 SAY "Press Enter to return to the menu"
15     @ 10,28 GET m_item
16     READ
17     @24,0
18     SEEK m_item
19     DO CASE
20     CASE m_item = SPACE(5)
21     <-----EXIT
22     CASE .NOT. FOUND()
23     | DO error WITH "Item No. " + TRIM(m_item) + " Not fond."
24     -----LOOP
25     CASE FOUND()
26     | @ 10, 28 GET item_no
27     | @ 12, 28 GET desc
28     | @ 14, 28 GET model
29     | @ 16, 28 GET locat
30     | @ 18, 28 GET onstock PICTURE "99999"
31     | @ 20, 28 GET price PICTURE "9,999.99"
32     | @ 20, 60 GET reorder PICTURE "99999"
33     CLEAR GETS
34     ENDCASE
35     * Determine if the operator wants to display another.
36     STORE " " TO m_ok
37     @ 24, 0
38     @ 24, 20 SAY "View another item record? (Y/N)----> " GET m_ok
39     READ
40     IF UPPER (m_ok) = "N"
41     <-----EXIT
42     ENDIF
43     @24,0
44 ENDDO
45 RETURN
46
47 *****
48 * End of file INVDIS.PRG *
49 *****

```

برنامج طباعة الأصناف (INVLST.PRG)

البرنامج الذي يقوم بطباعة التقرير الرئيسي للمخازن هو برنامج INVLST.PRG ويشتمل شكل ١٩ - ٥ على الخريطة التي توضح علاقة البرنامج بالنظام كما يشتمل شكل ٢٠ - ٥ على محتويات البرنامج.



شكل ١٩ - ٥ خريطة برنامج INVLST.PRG

```

1 * ----- *
2 * Program.....: invlst.prg *
3 * Called form.: invrep.prg *
4 * Called form.: invmenu.prg *
5 * Programmer...: Magdi M. Abu Al-Ata *
6 * Date.....: Dec. 25,1990 *
7 * Purpose.....: To display all inventory items *
8 * ----- *
9 PRIVATE scrttitle,scrfootr
10 scrttitle = "Inventory listing"
11 scrfootr = "Alphabetically by item description"
12 yn=" "
13 mor=" "
14 page = 0
15 row = 9
16 @ 24,5 SAY "P> Printer T> Terminal X> Exit to menu" GET yn PICTURE "I"
    
```

شكل ٢٠ - ٥ برنامج INVLST.PRG

```

17 DO intense WITH 24, 5,"P"      && Write P with different color
18 DO intense WITH 24,16,"T"      && Write T with different color
19 DO intense WITH 24,28,"X"      && Write X with different color
20 READ
21 DO CASE
22 CASE yn = "T"
23     DO border WITH scrttitle, scrfootr
24     DO WHILE .NOT. EOF()
25         page=page+1
26         @6,65 say "Page no. "+str(page,3)
27         @ row+1, 5 say "Item# Description Model Qty. Reorder"
28         @ row+2, 2 CLEAR TO 20,77
29         @ row+2, 5 say " " && put the cursor at the first line of report
30         DO WHILE ROW() < 20 .AND. .NOT. EOF() && The following are
31             && repeated lines
32             @ ROW()+1,5 SAY item_no
33             @ ROW(),COL()+1 SAY desc
34             @ ROW(),COL()+1 SAY model
35             @ ROW(),COL()+1 SAY onstock
36             @ ROW(),COL()+1 SAY reorder
37             SKIP
38         ENDDO
39         IF EOF()
40             @24,5 SAY "*** End of data. Press any key ***"
41             READ && To pause the program
42         ELSE
43             @24,5 SAY "More informations? [Y/N] " GET mor PICTURE "!"
44             READ
45             IF mor ="N"
46                 <-----EXIT
47             ENDIF
48         ENDIF
49     ENDDO
50 CASE yn ="P"
51     @ 24,5 SAY "Be sure your printer is ready and press ENTER "
52     READ
53     SET DEVICE TO PRINT
54     page = 1
55     pagetitle = "Alphabetical inventory listing"
56     fldtitle = "Item# Description Model Qty. Reorder"
57     cdate = str(DAY DATE(),2)+" "+CMONTH(DATE())+" "+STR(YEAR(DATE()),4)
58     DO WHILE .NOT. EOF()
59         @ 5, 5 SAY pagetitle
60         @ 5,65 SAY "Page No. "+ STR(page,3)
61         @ 6, 5 SAY "for: " + cdate
62         @ 9, 5 SAY fldtitle
63         @10, 5 SAY REPLICATE(CHR(95),LEN(fldtitle))
64         @11, 5 SAY " "
65         DO WHILE PROW() < 60 .AND. .NOT. EOF()
66             @ PROW()+1,5 SAY ITEM_NO
67             @ PROW(),PCOL()+1 SAY DESC
68             @ PROW(),PCOL()+1 SAY model
69             @ PROW(),PCOL()+1 SAY onstock
70             @ PROW(),PCOL()+1 SAY REORDER

```


- * يتم كتابة عناوين أعمدة التقرير في السطر التالي للسطر التاسع من الشاشة وهو أول سطر خال داخل الإطار (سطر ٢٧).
- * يتم مسح المنطقة التي ستطبع عليها محتويات التقرير (سطر ٢٨).
- * تنشأ دوائر جديدة داخل الدوائر الموجودة تستمر طالما أن رقم السطر الذي يطبع على الشاشة أقل من ٢٠ وطالما أن الملف لم ينته بعد (سطر ٣٠ - ٣٨) وقد استخدمنا في هذه الدوائر الوظيفة () ROW لتدل على رقم السطر الذي يقف عنده المؤشر والوظيفة () COL لتدل على موقع العمود الذي يقف عنده المؤشر وتلاحظ أننا نترك سطرًا خاليًا قبل طباعة محتويات التقرير ثم تكتب بيانات كل سجل في سطر واحد ويفصل بين أعمدة التقرير بمسافة واحدة.
- وبعد الانتهاء من طباعة بيانات السجل الأول ينتقل المؤشر داخل الملف إلى السجل الثاني (سطر ٣٧) . . . وهكذا فإذا وصل مؤشر الشاشة إلى السطر رقم ٢٠ يتم الخروج من الدائرة الداخلية. أي ينتقل التنفيذ إلى بداية الدائرة الخارجية. ويبدأ كتابة رقم الصفحة وعناوينها. . . وهكذا حتى ينتهي الملف.
- * أثناء طباعة التقرير يسأل البرنامج هل انتهت البيانات؟ فإذا كانت الإجابة بنعم يكتب رسالة تفيد انتهاء البيانات في السطر رقم ٢٤ (سطر ٣٩ - ٤١).
- أما إذا انتهت الصفحة ولم تنته بيانات الملف فتظهر رسالة تسأل المستفيد هل تريد الاستمرار في العرض أم لا (سطر ٤٣ - ٤٤). إذا قرر المستفيد الاكتفاء بما ظهر من بيانات يتم الخروج من الدائرة وإيقاف طباعة التقرير (سطر ٤٥ - ٤٧) وإلا تستمر طباعة التقرير حتى تنتهي بيانات الملف ويشتمل شكل ٢١ - ٥ على جزء من التقرير الذي سيظهر على الشاشة.

الحالة الثانية:

- إذا اختار المستفيد إرسال المخرجات إلى الطابعة يتم الآتي:
- لأن التقرير سيذهب إلى الطابعة فإذا كانت الطابعة غير مجهزة أو الورق غير راكب فستظهر رسالة أثناء التنفيذ تسبب إلغاء البرنامج لهذا يفضل في مثل هذه الحالة تنبيه المستفيد إلى تجهيز الطابعة تجنبًا لرسالة الخطأ ولكي يستمر البرنامج في التنفيذ

Sunday 3, March 1991
Time: 08:45:33

Arabian Computer Center
C.R. 39387

Page no. 1

Inventory listing

Item#	Description	Model	Qty.	Reorder
R-654	Floppy Disk	3M	3815	388
S-324	Hard Disk	CGAT	135	18
R-445	Modem	3338	7	3
S-319	Monitor	TVM	345	38
R-25C	PEGA Card		28	188
S-312	Printer	LQ588	4	5

Alphabetically by item description

** End of data. Press any key **

شكل ٢١ - هـ تقرير المخازن على الشاشة

- بشكل طبيعي . لذلك فإن البرنامج يظهر رسالة للتنبيه لهذا الغرض قبل بداية الطباعة (سطر ٥١ - ٥٢) .
- لأن مخرجات أمر SAY...@ تظهر تلقائياً على الشاشة ولأننا نريد توجيه هذه المخرجات إلى الطابعة استخدمنا أمر SET DEVICE (سطر ٥٣) .
- تنشأ حقول ذاكرة لتطبع بياناتها فيما بعد (سطر ٥٤ - ٥٧) ونتوقف أمام سطر ٥٧ لنوضح أن محتويات CDATE هي : الرقم الدال على تاريخ اليوم من التاريخ المخزن بالحاسب + فراغ + اسم الشهر الحالي + علامة الفاصلة + الرقم الدال على السنة الحالية وقد استخدمنا في هذا الأمر عدة وظائف هي () STR لتحويل العبارة الرقمية إلى حرفية والوظيفة () DAY لاستخراج اليوم المسجل بالتاريخ والوظيفة () DATE للدلالة على التاريخ المسجل بالحاسب والوظيفة CMONTH . () لاستخراج اسم الشهر الحالي والوظيفة () YEAR لاستخراج السنة الحالية بأربعة أرقام .
- تنشأ دوارية يتم الخروج منها إذا انتهت سجلات الملف (سطر ٥٨ - ٧٥) ومهمة .

الفصل السادس

المشتريات

Purchases

يشرح هذا الفصل نظام المشتريات وكيفية تسجيل فاتورة الشراء بالملف بعد التأكد من وجود اعتماد سابق لها واستخراج تقارير الشراء المطلوبة من فاتورة أو اعتماد.

والنظام يرتبط بنظام المخازن الذي شرعناه في الفصل السابق. ويتعامل مع ملف المخزون الرئيسي.

ويبدأ الفصل بشرح وظائف النظام ثم يشرح الملفات المطلوبة ثم يشرح بالتفصيل كل برنامج من برامج النظام على حدة وملائقته بغيره من البرامج الأخرى داخل النظام بحيث يمكنك فهم النظام وتوثيقه مع رغبات العميل إذا شئت.

قلنا أن المشتريات جزء من نظام المخازن ولا بد من فهم كل من المشتريات والمبيعات لفهم نظام المخازن وقد فصلنا كل من هذه الأنظمة في فصل مستقل من باب التيسير فقط.

والنظام الذي سنشرحه في هذا الفصل مهمته تسجيل البضاعة التي تصل من الاعتمادات المفتوحة على ملف متغيرات خاص بالمشتريات. وهذا الملف ملف تاريخي مهمته الاحتفاظ بالمشتريات لاستخراج التقارير التي تطلبها الإدارة فيما بعد مثل مشتريات فاتورة معينة أو اعتماد معين أو مشتريات خلال فترة معينة بالإضافة إلى أنه يستخدم لتعديل أرصدة الأصناف بالمخازن بعد عملية الشراء.

ولأن شركة الحاسب العربي تشتري البضاعة من الخارج باعتمادات مستندية فلا بد من تسجيل البضاعة بملف المشتريات بنفس رقم الاعتماد الذي سبق فتحه لدى البنك ولا بد أيضاً من تسجيل رقم الفاتورة. ولأن بضاعة الاعتماد الواحد قد تصل على أكثر من مرة وبالتالي على أكثر من فاتورة فيجب ألا تتشابه أرقام فواتير الشراء التي تخص نفس الاعتماد. ولذلك فعندما تصل البضاعة نسأل أولاً عن رقم الاعتماد الموجود بملف المشتريات - وبعد ذلك نسأل عن رقم الفاتورة داخل نفس الاعتماد. ويجب ألا يتكرر هذا الرقم داخل الاعتماد الواحد. فإذا لم يكن مكرراً أضيف أيضاً هذا الرقم إلى ملف المشتريات. ثم تسجل البضاعة الواردة بفاتورة الشراء بالملف أيضاً.

والنظام الذي سنشرحه في هذا الفصل يقوم بالوظائف التالية :

- ١ - تسجيل المشتريات من واقع فواتير الشراء الواردة بملف المشتريات .
PURCH.DBF ويتم تسجيل بضاعة الفاتورة في شكل مماثل لشكل فاتورة الشراء.
- ٢ - تقارير المشتريات وتشمل تقرير بمشتريات فاتورة أو تقرير بمشتريات اعتماد أو تقرير بمشتريات فترة معينة .

ويشتمل هذا النظام على برامج أكثر تعقيداً من برامج الفصول السابقة ويشتمل

على أفكار جديدة وهامة ولذلك يجب أن تكون فاهماً للفصول السابقة جيداً لكي يسهل عليك فهم هذا النظام.

وفيا يلي سنوضح الخطوات المطلوبة لتحقيق الوظائف السابقة.

تصميم ملفات قاعدة البيانات

- الملفات اللازمة للنظام الذي سنشرحه في هذا الفصل ثلاثة ملفات هي:
- ١ - ملف الاعتمادات (ORDER.DBF) وقد شرحناه في الفصل الرابع ونحتاج إليه للبحث فيه عن رقم الاعتماد قبل تسجيله بملف المشتريات. وأيضاً لمعرفة بيانات الاعتماد مثل اسم الشركة الموردة وعنوانها. . . إلخ.
 - ٢ - ملف المخزون الرئيسي (INV.DBF) وقد شرحناه في الفصل الخامس ونحتاج إليه للتحقق من وجود رقم الصنف قبل تسجيله ولمعرفة بيانات الصنف مثل اسمه ونوعه بدلاً من إعادة تسجيل هذه البيانات بملف المشتريات.
 - ٣ - ملف المشتريات (PURCH.DBF) وهو الملف التاريخي الذي سيشتمل على المشتريات ويتم تسجيل المشتريات بهذا الملف من واقع فواتير الشراء ونحتاج إليه لاستخراج التقارير اللازمة ولتعديل أرصدة المخزون الرئيسي بعد كل عملية شراء بإضافة الكميات المشتراة إلى الأرصدة الموجودة بالمخازن. ولإنشاء ملف PURCH.DBF أدخل الأمر الآتي من نقطة المحث

. CREATE PURCH

ثم أدخل المواصفات الموجودة بشكل ١ - ٦. ومنه يتضح أن ملف المشتريات يشتمل على البيانات التالية:

- ١ - رقم الاعتماد (ORD_NO) ويجب أن يشتمل على نفس المواصفات التي يشتمل عليها رقم الاعتماد في ملف الاعتمادات وهي نفس الاسم ونفس النوع ونفس الطول. وذلك لأننا سنستخدم هذا الرقم كمفتاح لربط الملفين فيما بعد.
- ٢ - رقم فاتورة الشراء (INVOICE_NO) وذلك لتمييز بضاعة كل فاتورة عن الأخرى.

Bytes remaining: 3957

Field Name	Type	Width	Dec	Field Name	Type	Width	Dec
1 ORD_NO	Character	5					
2 INVOICE_NO	Character	5					
3 ITEM_NO	Character	5					
4 QTY_RCV	Numeric	5	0				
5 PRICE	Numeric	7	2				
6 DATE	Date	8					
7 UPDATED	Logical	1					
8 B_DSCNT	Numeric	2	2				

MODIFY STRUCTURE [ESC] PURCH Field: 0-0

Enter the field name.

Field names begin with a letter and may contain letters, digits and underscores

شكل ١ - ٦ مواصفات ملف المشتريات

٣ - رقم الصنف (ITEM_NO). وهذا أيضاً يجب أن يخصص له نفس النوع ونفس الطول المخصص لرقم الصنف الموجود بملف المخزون الرئيسي لأننا سنستخدم هذا الرقم لربط الملفين معاً فيما بعد.

٤ - الكمية المستلمة (QTY_RCV)

٥ - السعر (PRICE).

٦ - تاريخ الشراء (DATE).

٧ - حقل يشتمل على علامة لتعرف منها هل هذه الكمية أضيفت بالفعل إلى رصيد المخزون الرئيسي أم لا وقد اخترنا أن يكون هذا الحقل حقل منطقي واختارنا له اسم UPDATED وتكون محتويات هذا الحقل F. بمجرد تسجيل الكمية وقبل إضافتها إلى رصيد المخازن. فإذا أضيفت إلى رصيد المخازن تعدلت إلى T. ويتولى برنامج مخصوص هذه المهمة وهذا البرنامج سنشرحه في الفصل التاسع إن شاء الله ضمن برامج صيانة الملفات.

٨ - مقدار الخصم (B_DSCNT) وهذا الحقل لتسجيل مقدار الخصم الممنوح على الفاتورة وقد أضفناه لحاجتنا إليه عند استرجاع بيانات فاتورة الشراء.

إنشاء ملفات الفهرسة

سنحتاج في هذا الفصل إلى أكثر من طريقة لترتيب الملف . وذلك تبعاً لحاجة البرامج والتقارير التي يتطلبها النظام . وهذا النظام يتطلب ترتيب ملف المشتريات بالطرق التالية :

- ١ - طبقاً لرقم الصنف لربط نظام المشتريات مع نظام المخزون الرئيسي فيما بعد .
 - ٢ - طبقاً لأرقام فواتير الشراء . وذلك للبحث عن فاتورة برقمها داخل الملف للتأكد من عدم تكرارها أثناء تسجيلها بالملف ولإظهار محتوياتها .
 - ٣ - طبقاً لرقم الاعتماد للبحث عن اعتماد برقمه داخل الملف لمعرفة الفاتورة أو الفواتير التي وصلت من هذا الاعتماد .
 - ٤ - طبقاً لرقم الاعتماد وفي داخل الاعتماد الواحد ترتب الفواتير التابعة له أيضاً للحاجة إلى إظهار أرقام الفواتير مرتبة داخل الاعتماد الواحد .
 - ٥ - طبقاً للترتيب رقم ٤ السابق بشرط ألا تتكرر الفواتير داخل نفس الاعتماد .
- ويوضح شكل ٢ - ٦ أسماء ملفات الفهرسة والحقول التابعة لها والمستخدمه في النظام .

ملف قاعدة البيانات	حقل الفهرسة	اسم ملف الفهرس	طريقة ترتيب الملف
Database file	Key	Inder file	Indexed By
PURCH	ITEM_NO	ITEM_P	طبقاً لرقم الصنف
	INVOICE_NO	PUR_INV	طبقاً لرقم الفاتورة
	ORD_NO	PUR_ORD	طبقاً لرقم الاعتماد
	ORD_NO+INVOICE_NO	ORDINV	طبقاً لرقم الاعتماد والفاتورة
	ORD_NO+INVOICE_NO	ORDINVU	طبقاً لرقم الاعتماد والفاتورة
			بدون تكرار لأرقام الفواتير داخل الاعتماد

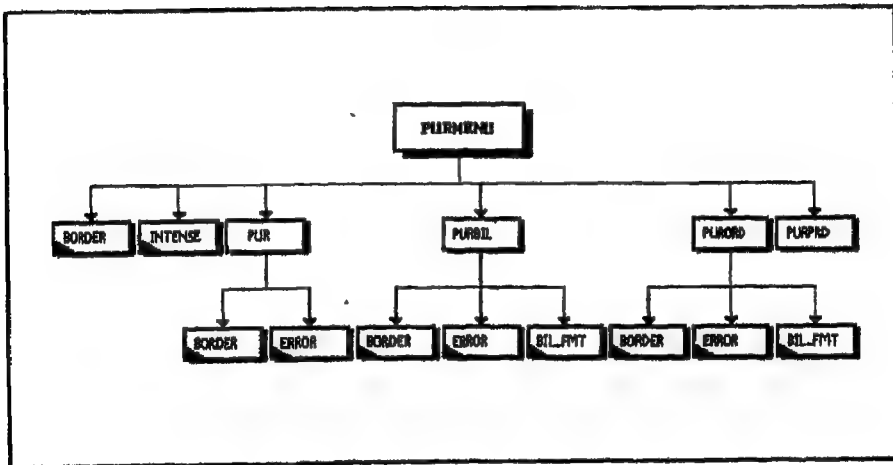
شكل ٢ - ٦ أسماء ملفات الفهرسة والحقول التابعة لها

ويجب أن تنشأ ملفات الفهرسة بعد الانتهاء من إنشاء ملف قاعدة البيانات وحفظ مواصفاته والأوامر اللازمة لإنشاء ملفات الفهرسة هي :

- . INDEX ON ITEM_NO TO ITEM_P
- . INDEX ON INVOICE_NO TO PUR_INV
- . INDEX ON ORD_NO TO PUR_ORD
- . INDEX ON ORD_NO + INVOICE_NO TO ORDINV
- . INDEX ON ORD_NO + INVOICE_NO TO ORDINV UNIQUE

تحديد برامج النظام

الخطوة التالية في تطوير النظام هي رسم خريطة للنظام توضح البرامج التي يشتمل عليها والتي تؤدي الوظائف المطلوبة من النظام . ويوضح شكل ٣ - ٦ خريطة النظام التي توضح العلاقة بين البرامج التي يشتمل عليها . ومن هذه الخريطة يتضح أن البرنامج الرئيسي وهو PURMENU.PRG يستدعي أربعة برامج أخرى كل منها يقوم بوظيفة محددة . ويمثل كل وظيفة من هذه الوظائف مستطيل داخل خريطة النظام بالإضافة إلى مجموعة إجراءات تعرفها بتظليل الركن اليسار السفلي من



شكل ٣ - ٦ خريطة نظام المشتريات

المستطيل . وهي كلها مألوفة لك من الأنظمة السابقة باستثناء إجراء واحد فقط جديد وهو BIL_FMT وستتناول فيما يلي شرح البرامج الموجودة بالخريطة بالإضافة إلى الإجراء الجديد.

قائمة نظام المشتريات

يجب أن تلمي قائمة المشتريات الوظائف الأساسية الموضحة بخريطة النظام وهي :

تسجيل مشتريات جديدة - إظهار مشتريات فاتورة - إظهار مشتريات اعتماد - إظهار مشتريات عن فترة محددة . ويوضح شكل ٤ - ٦ قائمة نظام المشتريات والاختيارات التي تشتمل عليها كما يشتمل شكل ٥ - ٦ على برنامج PURMENU.PRGM اللازم لإظهار هذه القائمة وتنفيذ اختياراتها وقد سبق أن شرحنا فكرة برامج القوائم في الفصول السابقة ولا تختلف فكرة هذا البرنامج عن برامج القوائم السابقة .

Tuesday 26, February 1991	Time: 15:17:03
Arabian Computer Center C.R. 39387	
Purchase menu	
(1) New purchases..... (2) Purchases by Bill (3) Purchases by Order..... (4) Purchases during a Period (G) Go to main menu..... (Q) Quit to DOS.....	
Select number or colored letter ==> █	

شكل ٤ - ٦ قائمة نظام المشتريات

```

1 * -----*
2 * Program.....: purmenu.prg*
3 * Called form.: dbapp.prg*
4 * Programmer...: Magdi M. Abu Al-Ata*
5 * Date.....: Dec. 25,1990*
6 * Purpose.....: To display purchase menu*
7 * -----*
8 PRIVATE scrtitle,scrfootr,choice
9 scrtitle = "purchase menu"
10 scrfootr = "Select number or colored letter ==>"
11 ok = .T.
12 DO WHILE ok
13     choice = " "
14     DO border WITH scrtitle,scrfootr
15     * Draw menu
16     @10, 5 SAY "(1) New purchases....."
17     DO intense WITH 10,10,"N"
18     @11, 5 SAY "(2) Purchases by Bill ....."
19     DO intense WITH 11,23,"B"
20     @12, 5 SAY "(3) Purchases by Order....."
21     DO intense WITH 12,23,"O"
22     @13, 5 SAY "(4) Purchases during a Period"
23     DO intense WITH 13,29,"P"
24     @14, 5 SAY "(G) Go to main menu....."
25     DO intense WITH 14,10,"G"
26     @15, 5 SAY "(Q) Quit to DOS....."
27     DO intense WITH 15,10,"Q"
28     @22,43 GET choice
29     READ
30     * Do selected option
31     DO CASE
32     CASE choice $ "1nN"                && If choice = 1 or n or N
33     | DO pur                            && Add new purchases
34     CASE choice $ "2bB"                && If choice = 2 or b or B
35     | DO purbil                        && View bill purchases
36     CASE choice $ "3oo"                && If choice = 3 or o or O
37     | DO purord                       && View order purchases
38     CASE choice $ "4pp"                && If choice = 4 or p or P
39     | DO purprd                       && View purchases during a period
40     CASE UPPER(choice) = "G"
41     | <-----RETURN                    && Go to Main menu
42     | CASE UPPER(choice) = "Q"
43     | | CLEAR
44     | <=====QUIT                    && Quit to DOS
45     | OTHERWISE
46     | <-----LOOP
47     | ENDCASE
48 ENDD
49 *****
50 * End of file PURMENU.PRG *
51 *****

```

تسجيل مشتريات جديدة

البرنامج اللزم لتسجيل المشتريات الجديدة عندما تصل فاتورة الشراء هو . PUR.PRГ وتتلخص فكرة البرنامج في أن البضاعة عندما تصل يجب أن نتأكد من وجود اعتماد سابق لها بملف الاعتمادات ORDER.DBF . فإذا وجد الاعتماد يتم إدخال رقم فاتورة الشراء ولا بد أن نتأكد أن رقم هذه الفاتورة لنفس الاعتماد غير موجود من قبل بملف المشتريات (PURCH.DBF) . وذلك لأن جهة الاعتماد الواحد واحدة، وهي لا تصدر فواتير مكررة. وإنما قد يتكرر رقم فاتورة من اعتماد آخر. فإذا وجد رقم الفاتورة بنفس الاعتماد فهذا يعني أن الفاتورة سجلت من قبل بملف المشتريات. أما إذا لم يوجد رقم الفاتورة فتظهر شاشة يتم تسجيل أصناف فاتورة الشراء بها ورغم أن كل صنف يحتل سجلاً مستقلاً داخل ملف المشتريات إلا أننا نخصص لكل صنف سطرًا واحدًا من الشاشة حتى يسهل على مدخل البيانات عمله.

ويوضح شكل ٦ - ٦ وشكل ٦ - ٧ كيفية تسجيل رقم الاعتماد ورقم الفاتورة. أما شكل ٦ - ٨ فيوضح شاشة تسجيل المشتريات قبل إدخال أصناف الشراء وتلاحظ أنها تشتمل على معلومات عن الاعتماد ورقم الفاتورة. وكذلك يشتمل شكل ٦ - ٩ على شكل الشاشة بعد الانتهاء من تسجيل بيانات الفاتورة وتنتج الأشكال من ٦ - ٦ إلى ٦ - ٩ من برنامج تسجيل المشتريات PUR.PRГ ويوضح شكل ١٠ - ٦ الخريطة التي تبين علاقة هذا البرنامج بغيره من البرامج داخل النظام. أما البرنامج نفسه فتجده في شكل ١١ - ٦ وأمام هذا البرنامج لنا وقفة لتوضيح فكرته بصفة إجمالية قبل شرح الأوامر والأفكار الجديدة التي يشتمل عليها.

يشتمل البرنامج على ٣ دورات متداخلة وفي كل دورة أكثر من اختيار وهذه الدورات هي:

١ - الدورة الأولى تتكرر حتى يدخل المستفيد فراغات محل رقم الاعتماد أو يضغط مفتاح الإدخال وبها ٣ احتمالات هي:

الاحتمال الأول: أن يدخل المستفيد فراغ ليخرج من الدورة وينهي البرنامج.

Tuesday 26, February 1991	Time: 15:28:35
Arabian Computer Center C.R. 39387	
Recording purchased items	
Order no. [REDACTED]	
Enter order no. or press ← to exit.	

شكل ٦ - ٦ إدخال رقم الاعتماد

Tuesday 26, February 1991	Time: 15:23:47
Arabian Computer Center C.R. 39387	
Recording purchased items	
Invoice no. [REDACTED]	
Enter invoice no. or press ← to exit.	

شكل ٧ - ٦ إدخال رقم فاتورة الشراء

الفصل السادس : نظام المشتريات

Invoice number : 334	Order number : A18-5
Company name : Capag	
Company address : America	
Recording date : 23/02/91	

Item no	Description	Quantity	Price	Amount

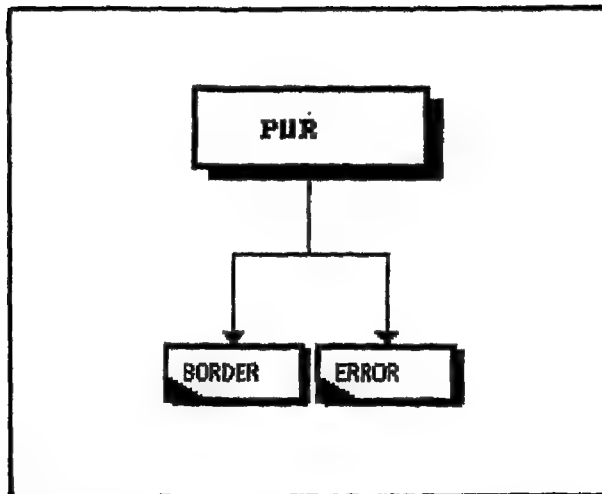
شكل ٨ - ٦ شاشة تسجيل المشتريات قبل إدخال أصناف الشراء

Invoice number : 334	Order number : A18-5
Company name : Capag	
Company address : America	
Recording date : 03/03/91	

Item no	Description	Quantity	Price	Amount
	PEGA Card			25,000.00
	Printer			18,000.00
	Modem			1,280.00
	Hard Disk			15,000.00
	Monitor			30,000.00
	Floppy Disk			1,500.00
	Total			90,780.00
	Discount			
	Final total			90,500.00

Correct data? ==> (Y/N) █

شكل ٩ - ٦ شكل الشاشة بعد انتهاء تسجيل المشتريات



شكل ١٠ - ٦ خريطة برنامج PUR.PRG

```

1 * ----- *
2 * Program.....: PUR.PRG *
3 * Called form.: PURMENU.PRG *
4 * Called form.: DBAPP.PRG *
5 * Programmer...: Magdi M. Abu Al-Ata *
6 * Date.....: Dec. 25,1990 *
7 * Purpose.....: To add goods received to purchase transaction file *
8 * ----- *
9 PRIVATE scrttitle,scrfootr,scfoot2,itemno,invoiceno,orderno
10 scrttitle = "Recording purchased items"
11 scrfootr = "Enter order no. or press " + ;
11 CHR(17) + CHR(196) + CHR(217) + " to exit."
12 scrfoot2 = " Enter invoice no. or press " + ;
12 CHR(17) + CHR(196) + CHR(217) + " to exit."
13 DO border WITH scrttitle,scrfootr
14 SELECT A
15 USE order
16 SELECT B
17 USE purch
18 SELECT C
19 USE inv
20 * Set up a loop to enter goods received
21 DO WHILE .T.
22   orderno = SPACE(5)
23   SELECT A
24   SET INDEX TO order      && Indexed on ord_no
25   @ 16,5 SAY " Order no " GET orderno PICTURE "@1"
  
```

شكل ١١ - ٦ برنامج PUR.PRG

```

26 READ
27 * Find entered order no. in order file.
28 SEEK orderno
29 DO CASE
30 CASE orderno = space(5)
31 <-----EXIT
32 * if order no. not found inform the user and try again
33 CASE .NOT. FOUND()
34 *send an error message
35 DO error WITH "Order no. " + rtrim(orderno) + " not found. "
36 CASE FOUND()
37 * Set a loop to accept bill(s).
38 DO WHILE .T.
39     invoiceno = SPACE(5)
40     DO border WITH scrttitle,scrfoot2
41     @ 16,5 SAY " Invoice no. " GET invoiceno PICTURE "@!"
42     READ
43     * Check if the order has same invoice no.
44     SELECT B
45     SET FILTER TO ord_no = orderno
46     SET INDEX TO pur_inv      && Indexed on invoice_no
47     GO TOP
48     SEEK invoiceno
49     DO CASE
50     CASE invoiceno = SPACE(5)
51     <-----EXIT
52     CASE FOUND()
53     DO error WITH "Invoice no. "+rtrim(invoiceno) + ;
54     " already exist. "
55     CASE .NOT. FOUND()
56     * Add bill data to purch.dbf
57     mtotals=0
58     CLEAR
59     @1,2 SAY "Invoice number : " + TRIM(invoiceno)
60     @1,45 SAY "Order number : " + TRIM(orderno)
61     @2,2 SAY "Company name : " + A->comp_nam
62     @3,2 SAY "Company address: " + A->comp_adr
63     @4,2 SAY "Recording date : " + DTOC(DATE())
64     @5,0 SAY REPLICATE(CHR(45),79)
65     ?? "      Item no      Description      Quantity"
66     ?? "      Price      Amount "
67     @7,0 SAY REPLICATE(CHR(45),79)
68     row=8
69     newlin=.T.
70     * Set a loop to accept the bill data
71     DO WHILE newlin
72     APPEND BLANK
73     REPLACE invoice_no WITH invoiceno
74     REPLACE date WITH date()
75     REPLACE ord_no WITH orderno
76     @row,0 GET item_no PICTURE "@!"
77     READ
78     STORE item_no to itemno && to use it with SEEK
79     * Make sure the item no. exist in INV.DBF file

```

```

79      itemno=trim(itemno)
80      SELECT C
81      SET INDEX TO item          && indexed on item_no
82      SEEK itemno
83      DO CASE
84      CASE len(itemno) = 0
85      |      newlin=.F.
86      |      CASE .not. FOUND()
87      |      DO error WITH " warning.. item no. not found"
88      |      row=row-1
89      |      CASE FOUND()
90      |      @row,11 SAY desc      && Say item description
91      |      SELECT B
92      |      @row,30 GET qty_rcv RANGE 1,
93      |      @row,43 GET price
94      |      read
95      |      @row,62 SAY qty_rcv*price PICTURE "9,999,999.99"
96      |      mttotal=mttotal+(qty_rcv*price)
97      |      ENDCASE
98      |      SELECT B      && In case itreno = spaces or not found
99      |      && INV.DBF is the current
100     |      row=row+1
101     |      * Scroll screen if neering end
102     |      IF row>19
103     |      |      @24,1
104     |      |      ?
105     |      |      row=19
106     |      ENDIF
107     |      ENDDO (while newlin = .t. )
108     |      @ row+1,42 SAY "Total "
109     |      @ row+1,64 SAY mttotal PICTURE "9,999,999.99"
110     |      * Calculate discount
111     |      redu = 0.00
112     |      @row+2,42 SAY "Discount"
113     |      @row+2,67 GET redu PICTURE "9,999,999.99"
114     |      read
115     |      fttotal=mttotal - redu
116     |      @row+3,63 SAY "-----"
117     |      @row+4,42 SAY "Final total"
118     |      @row+4,63 SAY fttotal PICTURE "9,999,999.99"
119     |      @row+5,63 SAY "=====
120     |      *
121     |      ok=" "
122     |      @ 24,1 SAY "Correct dta? ==> (Y/N)" GET ok PICTURE "!"
123     |      READ
124     |      @24,0
125     |      IF ok $ "YN"
126     |      |      IF ok= "N"
127     |      |      |      DELETE FOR invoice_no = invoiceno
128     |      |      |      CLEAR
129     |      |      |      LOOP
130     |      |      ENDIF
131     |      |      * Delete wrong records from purch.dbf
132     |      |      * that has invalid item no.

```

```
133      DELETE FOR qty_rcv=0
134      * Replace discount in purch.dbf
135      REPLACE b_dscnt WITH redu FOR invoice_no=invoice_no
136      ENDIF
137      ENDCASE
138      SET FILTER TO
139      ENDDO (while invoice_no # " ")
140      ENDCASE
141      ENDDO (WHILE order_no # " ")
142      CLOSE DATABASES
143      * Update inventory file (INV.DBF) from purchases file (PURCH.DBF)
144      * DO updtpr
145      RETURN
146      *****
147      * End of file PUR.PRG      *
148      *****
```

تابع شكل ١١ - ٦ برنامج PUR.PRG

الاحتمال الثاني: أن يكون رقم الاعتماد غير موجود وهذا يسبب ظهور رسالة خطأ والعودة إلى بداية الدوارة لتكرار إدخال الرقم الصحيح .
الاحتمال الثالث: أن يكون رقم الاعتماد موجوداً وهذا الاحتمال ينشئ الدوارة الثانية .

٢ - الدوارة الثانية تتكرر حتى يدخل المستخدم فراغات محل رقم الفاتورة وبها أيضاً ٣ احتمالات هي :

الاحتمال الأول: أن يدخل المستخدم فراغ يسبب الخروج من الدوارة والعودة إلى بداية الدوارة الأولى .

الاحتمال الثاني: أن يكون رقم الفاتورة مسجلاً من قبل وهذا يسبب ظهور رسالة خطأ والعودة إلى بداية الدوارة لتكرار إدخال الرقم الصحيح .
الاحتمال الثالث: أن يكون رقم الفاتورة غير موجود وهنا يتم قبول بيانات الفاتورة .

والآن نعود لشرح منطق البرنامج والأوامر التي يشتمل عليها بالتفصيل :
- يبدأ البرنامج بتخصيص حقول ذاكرة لتستخدم داخل هذا البرنامج فقط (سطر رقم ٩) .

- ينشئ حقول ذاكرة حرفية لتستخدم داخل البرنامج (سطر ١٠ - ١٢) وقد استخدمنا في سطر ١١، ١٢ الوظيفة () CHR لنحصل على الحرف المقابل للعدد وبها يمكن إظهار حروف غير موجودة بلوحة المفاتيح. والحروف الثلاثة المقابلة للأعداد ١٧، ١٩٦، ٢١٧ ينتج عنها هذه العلامة ←.
- يتم اختيار المنطقة الأولى (سطر ١٤) وفتح ملف order بها (سطر ١٥) لأن به أرقام الاعتمادات ونحن سنبحث عن رقم الاعتماد الذي أدخل في ملف الاعتمادات. وبالمثل نختار المنطقة الثانية (سطر ١٦) ونفتح ملف Purch بها (سطر ١٧) لأنه الملف الذي تسجل به المشتريات وكذلك نختار المنطقة الثالثة (سطر ١٨) ونفتح بها ملف inv (سطر ١٩) لأن به البيانات الأساسية للأصناف. ومعروف أننا يمكننا تقسيم الذاكرة إلى عشرة مناطق ليوضع بكل منطقة ملف قاعدة بيانات ونخص كل منطقة رقم من ١ إلى ١٠ أو حرف من A إلى J ليدل عليها. ويبقى المؤشر داخل أي ملف في موضعه داخل الملف إذا انتقلنا من منطقة (أو ملف) إلى أخرى.
- تنشأ دوائر لانهائية بأمر DO WHILE .T. في سطر ٢١ وهذه الدوائر تتكرر إلى ما لا نهاية إلا إذا تحقق الشرط الذي يستدعي أمر EXIT. ليتم الخروج منها.
- يتم اختيار المنطقة A. وبها ملف ORDER.DBF ليكون هو الملف الحالي (سطر ٢٣) ويتم اختيار الترتيب طبقاً لرقم الاعتماد (سطر ٢٤).
- يتم قبول رقم الاعتماد من المستفيد (سطر ٢٥، ٢٦) والبحث عن هذا الرقم داخل الملف المفتوح (سطر ٢٨) وهنا يمكن أن نتحقق واحدة من الحالات التالية:
- الحالة الأولى أن تدخل فراغات أي أن يضغط المستفيد مفتاح الإدخال (سطر ٣٠) ونتيجتها الخروج من الدائرة أي إنهاء البرنامج.
- الحالة الثانية أن يكون رقم الاعتماد غير موجود (سطر ٣٣) وهنا تظهر رسالة تفيد ذلك وتطلب إعادة المحاولة لأن معنى ذلك أن المستفيد أخطأ في كتابة الرقم وبالتالي يتكرر تنفيذ الدائرة (سطر ٣٥).
- الحالة الثالثة أن يكون رقم الاعتماد موجوداً (سطر ٣٦) وفي هذه الحالة تفتح دوائر داخل الدائرة الأولى ليدخل من خلالها رقم الفاتورة. وبعد إدخال رقم الفاتورة

- يُختار ملف المنطقة B وهو ملف PURCH.DBF (سطر ٤٤).
- يتم اختيار السجلات التي تخص رقم الاعتماد الذي أدخل في الدوارة الأولى فقط (سطر ٤٥) وتهمل باقي السجلات من وجهة نظر قاعدة البيانات.
 - يتم اختيار فهرس PUR_INV.NDX (سطر ٤٦) ويعني ترتيب الملف طبقاً لرقم الفاتورة لأننا سنبحث فيه عن رقم الفاتورة الذي أدخل.
 - نبحث عن رقم الفاتورة في ملف المشتريات PURCH.DBF (سطر ٤٨) وبعد البحث يتم تنفيذ واحدة من الثلاث حالات التالية:
- ١ - الخروج من الدوارة إذا لم يدخل شيء محل رقم الفاتورة (سطر ٥٠ - ٥١).
 - ٢ - إذا كان الرقم موجوداً يتم إعلام المستفيد بأن الرقم موجود داخل نفس الاعتماد وإعطائه فرصة لتكرار إدخال الرقم الصحيح (سطر ٥٢ - ٥٣).
 - ٣ - إذا كان الرقم غير موجود (سطر ٥٤) يحصل الآتي:
- * إنشاء حقل ذاكرة رقمي (سطر ٥٦).
- * مسح الشاشة (سطر ٥٧) وكتابة البيانات التالية في السطور الأربعة الأولى: رقم الفاتورة (سطر ٥٨)، ورقم الاعتماد (سطر ٥٩)، واسم الشركة الموردة ويؤخذ من الملف الموجود بالمنطقة A وهو ملف الاعتمادات ORDER.DBF (سطر ٦٠). ولذلك فإن هذه العلامة > - A تعني البحث عن حقل Comp_nam في المنطقة A. وعنوان الشركة (سطر ٦١) وتاريخ اليوم (سطر ٦٢) وقد استخدمنا الوظيفة () DTOC لأن العبارة حرفية.
- في سطر ٦٣ يتم تكرار الحرف المقابل للكود ٤٥ عدد ٧٩ مرة وهذا الحرف هو علامة الشرطة.
 - تكتب عناوين الحقول في السطر التالي - لاحظ أننا استخدمنا أمر ?? لتكتب البيانات التي تليه في نفس السطر.
 - تنشأ دوارة لإضافة سجل جديد تتكرر حتى تصير محتويات newlin هي F. (سطر ٧٠) وتبدأ الدوارة بإضافة سجل خالٍ في نهاية الملف المفتوح وهو ملف المشتريات (سطر ٧١) واستبدال حقل رقم الفاتورة بالرقم الذي أدخله المستفيد من لوحة

- المفاتيح (سطر ٧٢)، وحقل التاريخ بالتاريخ المسجل بالحاسب (سطر ٧٣)، ورقم الاعتماد بالرقم الذي أدخل (سطر ٧٤).
- يتم قبول رقم الصنف وتخزينه بحقل آخر لأننا سنبحث عن هذا الرقم في ملف المخزون الرئيسي INV.DBF.
- نختار المنطقة C ليكون ملف INV.DBF هو الملف الحالي (سطر ٨٠) ونختار معه الترتيب طبقاً لرقم الصنف (سطر ٨١).
- نبحث عن رقم الصنف الذي أدخل في ملف المخزون وهنا أيضاً أمامنا ٣ احتمالات:
- * أن ندخل فراغ محل رقم الصنف وهذا معناه أن طول رقم الصنف = صفر لأننا طبقاً لم ندخل شيئاً (سطر ٨٤) وهذا يسبب استبدال محتويات newlin بالقيمة F. أي الخروج من الدوارة والانتقال إلى الدوارة السابقة (سطر ٨٥).
- * أن يكون الرقم الذي أدخل غير موجود بالملف وهنا تظهر رسالة خطأ ويُعطى المستفيد الفرصة لتكرار إدخال الرقم الصحيح (سطر ٨٦ - ٨٧).
- في حالة إدخال رقم صنف خطأ نظرح ١ من رقم السطر الحالي لأننا دائماً نضيف ١ لرقم السطر للانتقال إلى سطر جديد لإدخال الصنف التالي من الفاتورة. وفي هذه الحالة نود إدخال رقم الصنف الصحيح في نفس السطر الذي أدخل فيه الرقم الخطأ (سطر ٨٨).
- * أن يكون الرقم موجوداً بملف المخازن (سطر ٨٩) فيتم كتابة اسم الصنف لأن اسم الصنف مسجل بملف المخازن (سطر ٩٠) ثم نختار ملف المشتريات ثانية (سطر ٩١) ويتم قبول الكمية والسعر في السجل الحالي الذي أدخلناه قبل ذلك (سطر ٩٢ - ٩٤). نضرب الكمية \times السعر ونكتب الناتج على الشاشة (سطر ٩٥) ثم يتم تجميع حاصل ضرب الكمية \times السعر تمهيداً لمعرفة إجمالي الفاتورة. (سطر ٩٨).
- نعود لاختيار ملف المشتريات (سطر ٩٨) لأن ملف المخزون سيبقى هو الملف الحالي في حالتين: إذا أدخلنا فراغات محل رقم الصنف أو إذا كان رقم الصنف غير موجود بالمخازن.

- نضيف ١ لرقم السطر الحالي لننقل المؤشر إلى السطر التالي لموقعه على الشاشة (سطر ١٠٠).

- لأننا لا ندرى عدد الأصناف التي ستدخل من فاتورة الشراء فهناك احتمال أن تزيد عن المساحة المخصصة على الشاشة ولذلك فإذا قاربنا من نهاية الشاشة فإننا نلجأ إلى حيلة صغيرة وهي طي الشاشة سطر لأعلى إذا وصلنا إلى سطر رقم ١٩ على الشاشة (سطر ١٠٢ - ١٠٥).

- قلنا إن هذه الدوارة تغلق إذا أدخلنا فراغات محل رقم الصنف وهذا يحدث إذا انتهت الأصناف التي نريد تسجيلها فإذا حدث ذلك فإننا نسجل المجموع والخصم والمجموع النهائي للفاتورة. (سطر ١٠٨ - ١١٩). وقد اخترنا الأماكن المناسبة التي ستظهر أمام المجموع مثل كلمة Total أو الخصم مثل كلمة Discount

- يعطي البرنامج فرصة أخرى لإلغاء هذه الفاتورة قبل تسجيلها إذا رأى المستفيد أنها تشتمل على بيانات خاطئة (سطر ١٢١ - ١٣٠) فإذا رد المستفيد على رسالة هل البيانات صحيحة بلا (N) يتم حذف السجلات التي تشتمل على رقم الفاتورة التي أدخلت (سطر ١٢٧) وهذه هي الوسيلة التي تميز سجلات هذه الفاتورة عن غيرها من سجلات الملف ثم تمحي الشاشة (سطر ١٢٨) ونعود لبداية الدوارة الحالية (سطر ١٢٩) أي الدوارة التي تطلب إدخال رقم فاتورة (لاحظ اتجاه السهم الموجود أمام أمر Loop).

- قلنا أن المستفيد إذا أخطأ في إدخال رقم الصنف تظهر رسالة تفيد ذلك ويعود لتكرار إدخال الرقم الصحيح إلا أن الرقم الخطأ تم إضافته إلى ملف PURCH.DBF لأننا نضيف دائماً سجلاً خالياً قبل كتابة رقم الصنف ولذلك فإننا نلجأ قبل إنهاء البرنامج إلى حذف هذه السجلات الخطأ. ونعرف هذه السجلات الخطأ لأنها تشتمل فقط على رقم السجل وبالتالي فإنها ستشتمل على الرقم صفر في حقل الكمية (سطر ١٣٣). لاحظ أن أمر DELETE في سطر ١٢٧ وسطر ١٣٣ يضع علامة فقط أمام السجلات أما الحذف النهائي فيتم بأمر PACK. وقد استبعدنا أمر PACK في هذا البرنامج لأنه يستغرق وقتاً طويلاً أثناء التنفيذ ولأننا

- خصصنا وظيفة لهذا الغرض ضمن برامج صيانة الملفات في الفصل التاسع.
- يتم استبدال حقل B_DSCNT بقيمة خصم الفاتورة - إن وجد - لاسترجاعه فيما بعد في برامج التقارير (سطر ١٣٥).
- قبل إدخال بيانات فاتورة جديدة نعيد ملف المشتريات إلى حالته السابقة بأمر SET FILTER TO (سطر ١٣٨).
- بعد إغلاق الملفات وقبل الخروج من البرنامج تلاحظ أننا وضعنا علامة * أمام أمر DO updtpu (سطر ١٤٤) لإلغاء عمل هذا الأمر. وقد عمدنا إلى ذلك بدلاً من حذف هذا السطر والغاية من ذلك توضيح أنه بإمكانك تنفيذ برنامج تعديل أرصدة المخازن بناء على كميات المشتريات التي تم إضافتها. أو تأجيل ذلك لتنفيذ هذا البرنامج فيما بعد ضمن برامج صيانة الملفات. وقد اخترنا الطريقة الأخيرة وهي تأخير تعديل أرصدة المخازن الآن والقيام بتعديلها ببرنامج خاص ضمن برامج صيانة الملفات وسيأتي شرحه في الفصل التاسع إن شاء الله.

الاستفسار عن فاتورة شراء.

الاختيار الثاني في قائمة المشتريات هو الاستفسار عن مشتريات فاتورة (راجع شكل ٤ - ٦) وفي هذا البرنامج يحتاج مدير الشركة عند الاستفسار عن فاتورة الشراء لإظهار محتويات الفاتورة بشكل مماثل لشكل الفاتورة الواردة ويوضح شكل ١٢ - ٦ شكل فاتورة تم الاستفسار عنها بإدخال رقمها والبرنامج اللازم للاستفسار عن فاتورة شراء هو PURBIL.PRG وتتلخص وظيفته في التأكد من وجود رقم الفاتورة أولاً. ثم إظهار بياناتها والخصم إن وجد والإجمالي النهائي. ويشتمل شكل ١٣ - ٦ على الخريطة التي توضح علاقة هذا البرنامج بغيره من البرامج داخل النظام كما يشتمل شكل ١٤ - ٦ على محتويات البرنامج وعن هذا البرنامج نوضح ما يلي:

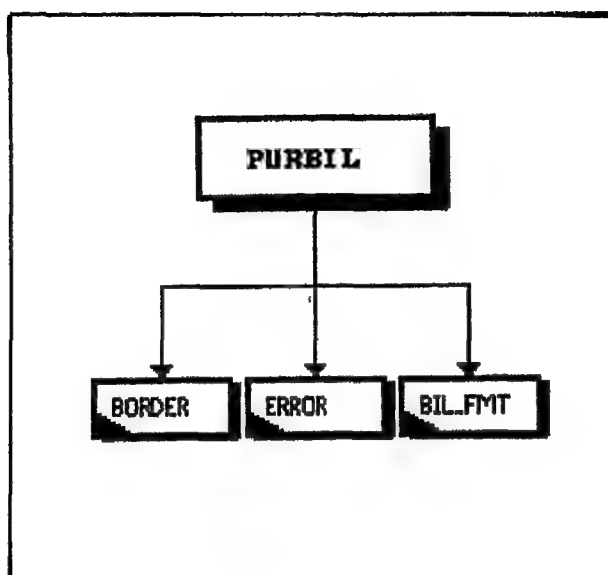
يستدعي البرنامج ٣ إجراءات منها إجراء واحد جديد وهو BIL_FMT سنشرحه بعد الانتهاء من هذا البرنامج. والبرنامج أيضاً يستخدم مفهوم التعامل مع أكثر من ملف موجود بالذاكرة ويستخدم أمر SELECT الذي شرحناه في البرنامج

الفصل السادس: نظام المشتريات

Invoice No. 3334 =====				
Item No.	Description	Quantity	Price	Amount
R-25C	PEGA Card	50	900.00	45000.00
S-312	Printer	10	1800.00	18000.00
R-445	Modem	3	400.00	1200.00
S-324	Hard Disk	20	950.00	19000.00
S-319	Monitor	50	700.00	35000.00
R-654	Floppy Disk	1000	2.00	2000.00
Total invoice				120200.00
Discount				200.00
Final total				120000.00 =====

Inquire another invoice? [Y/N] █

شكل ١٢ - ٦ إظهار بيانات فاتورة شراء



شكل ١٣ - ٦ خريطة برنامج PURBIL.PRG

```

1 * ----- *
2 * Program.....: purbil.prg *
3 * Called form.: purmenu.prg *
4 * Called form.: dbapp.prg *
5 * Programmer...: Magdi M. Abu Al-Ata *
6 * Date.....: Dec. 25,1990 *
7 * Purpose.....: To display purchases for bill of loading *
8 * ----- *
9 PRIVATE scrtitle,scrfootr,invno
10 scrtitle = "Displaying invoice purchases"
11 scrfootr = "Enter invoice no. or press Enter ";
11 CHR(17)+CHR(196)+CHR(217)+ " key to exit "
12 yn = " "
13 invno = "?"
14 fttotal = 0.00
15 SELECT B
16 USE inv INDEX item          && Indexed on item_no
17 SELECT A
18 USE purch
19 DO WHILE invno # " "
20     SET INDEX TO pur_inv          && Indexed on invoice_no
21     DO border with scrtitle,scrfootr
22     * Get wanted invoice no.
23     invno = SPACE(5)
24     @ 16,5 SAY " Invoice no. " GET invno PICT "@!"
25     READ
26     * Check if the invoice already exist
27     SEEK invno
28     DO CASE
29     CASE invno = " "
30     <-----EXIT
31     CASE .NOT. FOUND()
32     DO error WITH "Invoice no. " + rtrim(invno) + " not found "
33     CASE FOUND()
34     SET INDEX TO          && Close index to set filter
35     SET HEAD OFF          && Turn field title off
36     SET FILTER TO invoice_no = invno
37     GO TOP
38     SET RELATION TO item_no INTO B
39     DO bil_fmt
40     SET HEAD ON
41     SET FILTER TO
42     *
43     @ 24,5 SAY "Inquire another invoice? [Y/N] " GET YN PICT "!"
44     READ
45     IF yn='N'
46     <-----EXIT
47     ENDIF
48     ENDCASE
49 ENDDO          && While invoice no # spaces
50 CLOSE DATABASES
51 RETURN
52 *****
53 * End of file PURBIL.PRG *
54 *****

```

- السابق. وينشئ دوائر تتكرر حتى يدخل المستفيد فراغات محل رقم الفاتورة (سطر ١٩ - ٤٩). أو يرد على رسالة تظهر بعد عرض محتويات الفاتورة بلا (N) وتبدأ الدائرة بقبول رقم الفاتورة من المستفيد (سطر ٢٤ - ٢٥) والبحث عنها في ملف المشتريات (سطر ٢٧). ثم تنفيذ واحدة من الحالات التالية:
- الخروج من الدائرة نتيجة لتنفيذ أمر EXIT بعد إدخال فراغات محل رقم الفاتورة (سطر ٢٩ - ٣٠).
 - إظهار رسالة خطأ إذا لم تكن الفاتورة موجودة بالملف ثم إعادة المحاولة (سطر ٣١ - ٣٢).
 - إظهار محتويات الفاتورة إذا كانت موجودة بالملف (٣٣ - ٣٩) ولكي تظهر محتويات فاتورة الشراء على شاشة واحدة يتم الآتي:
- ✱ إغلاق ملف الفهرس (سطر ٣٤) وإلغاء أسماء الحقول التي تظهر فوق البيانات مع أمر LIST أو DISPLAY (سطر ٣٥).
 - ✱ جعل ملف المشتريات يبدو كما لو كان يشتمل على سجلات الفاتورة المقصودة فقط باستخدام أمر SET FILTER (سطر ٣٦).
 - ✱ ربط ملف المشتريات مع ملف المخازن الموجودة في المنطقة B مرتباً طبقاً لرقم الصنف. ربطها معاً باستخدام الحقل المشترك بينهما وهو رقم الصنف. (ITEM_NO) (سطر ٣٨) والغرض من ذلك أن تظهر بيانات السجل الذي يحمل نفس رقم الصنف من الملفين في سطر واحد كما سنرى داخل الإجراء BIL_FMT بعد قليل.
 - ✱ استدعاء الإجراء BIL_FMT لكتابة محتويات الفاتورة (سطر ٣٩).
- بعد إظهار محتويات الفاتورة يتم إعادة أمر SET HEAD إلى وضعه الأصلي (سطر ٤٠) وكذلك إعادة الملف إلى وضعه الأصلي الذي يظهر كل سجلات الملف (سطر ٤١).
- ثم يسأل المستفيد عما إذا كان يريد الاستفسار عن فاتورة أخرى أم لا. فإذا كانت الإجابة لا يتم الخروج من الدائرة وإنهاء البرنامج.

الإجراء BIL_FMT

يشتمل شكل ١٥ - ٦ على محتويات الإجراء BIL_FMT وتتلخص وظيفته في إظهار محتويات الفاتورة على الشاشة وقد وضعنا الأوامر التي تظهر محتويات الفاتورة ضمن ملف الإجراءات في إجراء مستقل بالرغم من إمكانية وضعها داخل البرنامج

```

240 *****
241 * Procedure : bil_fmt
242 * Purpose : Drows bill(s) screen format
243 *****

244 PROCEDURE bil_fmt
245 CLEAR
246 @ 0,5 SAY "Invoice No. "+RTRIM(invno)
247 @ 1,5 SAY "===== "+ REPLICATE(CHR(61),LEN(TRIM(invno)))
248 @ 3,0 SAY REPLICATE(CHR(196),80)
249 @ 4,0 SAY " Item No. Description"+SPACE(9)+"Quantity"+ ;
249 SPACE(6)+"Price"+SPACE(18)+"Amount"
250 @5,0 SAY REPLICATE(CHR(196),80)
251 LIST OFF item no,SPACE(3),B->desc,SPACE(5),qty_rcv,SPACE(7), ;
251 price,SPACE(12),qty_rcv * price
252 SUM qty_rcv*price TO bttotal
253 @ ROW()+1,64 SAY "-----"
254 @ ROW()+1,40 SAY "Total invoice"
255 @ ROW(),66 SAY bttotal
256 SKIP - 1
257 @ROW()+1,40 SAY "Discount"
258 @ROW(),72 SAY b_dscnt
259 fttotal=bttotal - b_dscnt
260 @ ROW()+1,64 SAY "-----"
261 @ROW()+1,40 SAY "Final total"
262 @ROW(),66 SAY FTOTAL
263 @ ROW()+1,64 SAY "===== "
264 RETURN
265
266 *****
267 * Procedure : pswrd
268 * Purpose :
269 *****

```

شكل ١٥ - ٦ الإجراء bil_fmt

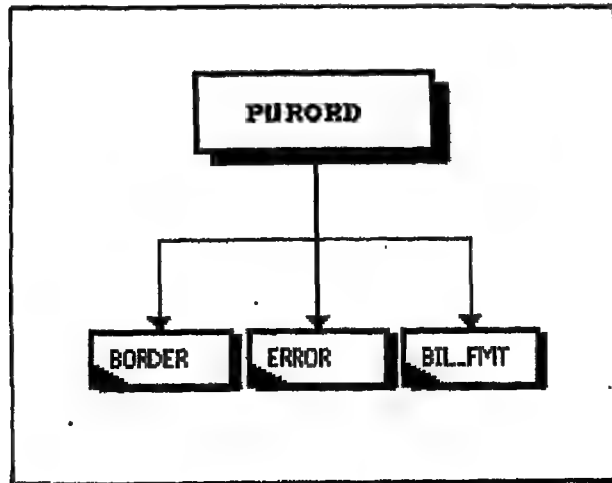
السابق لأن البرنامج السابق وبرنامج إظهار محتويات الاعتماد يستخدم نفس الأوامر وقد أردنا تجنب تكرار كتابتها مرتين. والإجراء يبدأ بمسح الشاشة (سطر ٢٤٥)

- وكتابة رقم الفاتورة (سطر ٢٤٦) وأسماء أعمدة التقدير (سطر ٢٤٨ - ٢٥٠) وقد استخدمنا لهذا الغرض الوظيفة () TRIM لحذف الفراغات الموجودة على يمين الرقم إن وجدت والوظيفة () REPLICATE لتكرار حرف الشرطة المتصلة لرسم خط أفقي بعرض الشاشة والوظيفة () LEN وتعني طول كذا.
- استخدمنا أمر LIST (سطر ٢٥١) وهو من الأوامر البدائية بعد تطويره ليعمل بكفاءة أكثر وذلك بإضافة الاختيار OFF ويعني إلغاء أرقام السجلات التي تظهر على يسار السجل تلقائياً وترك فراغات مناسبة بين الحقول لتناسب العناوين التي اخترناها في سطر ٢٤٩ وفي آخر حقل استخرجنا حاصل ضرب الكمية (QTY_RCV) × السعر (PRICE). وعدد الفراغات بين الحقول في أمر LIST مسألة تنظيمية ترجع إليك حسب مساحة الشاشة ومواقع البيانات عليها.
 - استخرجنا مجموع حاصل ضرب الكمية × السعر لنحصل على إجمالي الفاتورة قبل الخصم (سطر ٢٥٢) وقد وضعنا الناتج في حقل الذاكرة BTOTAL لإظهاره بعد ذلك.
 - كتبنا إجمالي الفاتورة في الموقع المناسب على الشاشة (سطر ٢٥٤ - ٢٥٥).
 - رجعنا بال مؤشر الموجود بالملف سجلاً للخلف (سطر ٢٥٦) لأن أمر LIST يضع المؤشر في آخر الملف ورجوعنا إلى الخلف بمقدار سجل واحد يتيح لنا قراءة مبلغ الخصم الموجود ضمن حقول السجل.
 - كتبنا الخصم في الموقع المناسب على الشاشة (سطر ٢٥٧ - ٢٥٨).
 - بطرح الخصم من إجمالي الفاتورة تحصل على الصافي النهائي للفاتورة (سطر ٢٥٩).
 - كتبنا الإجمالي النهائي في الموقع المناسب على الشاشة بعد رسم خطوط مناسبة قبله وبعده (سطر ٢٦٠ - ٢٦٣).

الاستفسار عن مشتريات اعتماد

يتيح الاختيار الثالث في قائمة المشتريات الاستفسار عن مشتريات اعتماد (راجع شكل ٤ - ٦ السابق) لأن الاعتماد قد يصل على فاتورة أو أكثر كما سبق أن

أوضحنا. والبرنامج اللازم لهذا الغرض هو PURORD.PRG. ويشتمل شكل ١٦ - ٦ على الخريطة التي توضح علاقة هذا البرنامج بغيره من البرامج داخل النظام. وهو يستدعي الإجراءات التي يستدعيها برنامج الاستفسار عن مشتريات فاتورة السابق. كما يشتمل شكل ١٧ - ٦ على محتويات هذا البرنامج ونلخص فكرة هذا



شكل ١٦ - ٦ خريطة برنامج PURORD.PRG

```

1 * -----*
2 * Program   : PURORD.PRG                      *
3 * Author    : Magdi M. Abu Al-Ata             *
4 * Date      : October 24, 1990                 *
5 * Purpose   : Displays purchases by one order  *
6 * Called from: PURMENU.PRG                     *
7 * Called from: DPAPP.PRG                       *
8 * -----*
9 PRIVATE scrtitle,scrfootr,ordno,invno
10 scrtitle = "Displaying order purchases"
11 scrfootr = "Enter order no. or press Enter "+ ;
12   CHR(17)+CHR(196)+CHR(217)+ " key to exit "
13 invno = SPACE(5)
14 mor = " "
15 SELECT B
16 USE inv INDEX item      && Indexed on item_no
17 SELECT A
18 USE purch
    
```

شكل ١٧ - ٦ برنامج PURORD.PRG

```

18 ordno = "?"
19 DO WHILE ordno # " "
20     DO border WITH scrttitle,scrfootr
21     ordno = SPACE(5)
22     @ 16,5 SAY " Order no. " GET ordno PICTURE "@!"
23     READ
24     * Check if the order already exist
25     SET INDEX TO pur_ord      && Indexed on ord_no
26     SEEK ordno
27     DO CASE
28     CASE ordno = " "
29     <-----EXIT
30     CASE .NOT. FOUND()
31     * No search required. a message to inform the operator
32     DO error with " the order " + RTRIM(ordno) + " not found "
33     CASE FOUND()
34     DO WHILE .T.
35         SET INDEX TO ordinvu      && On ord_no+invoice_no UNIQUE
36         CLEAR
37         @1,0 SAY "Bill(s) belong to order no. " +ordno
38         @2,0 SAY "===== " + ;
38         REPLICATE(CHR(61),LEN(RTRIM(ordno)))
39         ?
40         * Display invno(s) belong to it
41         SET HEAD OFF
42         LIST OFF invoice_no FOR ord_no = ordno
43         @ 24,5 SAY "Enter requested invno no. to display" ;
43         GET invno PICTURE "@!"
44         READ
45         SET INDEX TO pur_inv      && Indexed on invoice_no
46         SEEK invno
47         IF FOUND()
48             mdsct=b_dscnt
49             SET RELATION TO item_no INTO B
50             SET FILTER TO invoice_no=invno .AND. ord_no=ordno
51             DO bil_fmt
52             invno=SPACE(5)
53             @24,2 SAY "Do you need more bills? [Y/N] " ;
53             GET mor PICTURE "!"
54             READ
55             IF mor = "N"
56             <-----EXIT
57             ENDIF
58         ENDIF
59     ENDDO
60     SET HEAD ON
61     ENDCASE (order no found)
62     SET FILTER TO      && Reset PURCH.DBF to its normal records
63 ENDDO (while order no # " "
64 CLOSE DATABASES
65 RETURN
66
67 *****
68 * End of file PURORD.PRG *
69 *****

```

البرنامج كما يلي :

هذا البرنامج قريب من البرنامج السابق PURBIL.PRГ ويزيد عليه أننا نسال أولاً عن رقم الاعتماد. فإذا كان موجوداً طلبنا إظهار أرقام كل الفواتير التي تخصه ليختار منها المستفيد الفاتورة التي يريدّها. فإذا اختار فاتورة معينة ظهرت بياناتها بنفس الطريقة التي تظهر بها بيانات الفاتورة التي شرحناها في البرنامج السابق لأنه يستخدم نفس الإجراء الذي يظهر بيانات الفاتورة وهو bil_fmt ويتكرر سؤال المستفيد عن رقم الفاتورة المطلوبة حتى يقرر إنهاء البرنامج بالرد على الرسالة بلا.

والبرنامج يشتمل على دوائر خارجية لتكرار إدخال رقم الاعتماد (سطر ١٩ - ٦٣) ودوائر داخلية لتكرار إدخال رقم الفاتورة المطلوبة (سطر ٣٤ - ٥٩).

وسنناقش فيما يلي الأوامر الجديدة والتي لم ترد بالبرنامج السابق PURBIL.PRГ فقط.

- إذا وجد البرنامج رقم الاعتماد بملف المشتريات يتم فتح الفهرس ORDINVU.NDX (سطر ٣٥) وهذا الفهرس يرتب الملف طبقاً لرقم الاعتماد وفي داخل الاعتماد الواحد يرتب الفواتير أيضاً. وكما تذكر أننا أضفنا مع أمر INDEX عند فهرسة الملف الاختيار UNIQUE ويعني حذف السجلات المكررة من الملف عند إظهار محتوياته. وكما تعلم أن الفاتورة الواحدة يتكرر رقمها مع كل السجلات التي تخصها. ولا نرى ضرورة لتكرار هذا الرقم. وإنما يكفي أن يظهر الرقم مرة واحدة لأننا سنختاره لإظهار محتويات الفاتورة بشكل منظم بعد ذلك.

- استخدمنا أمر LIST (سطر ٤٢) لإظهار أرقام الفواتير التي تخص الاعتماد الذي أدخل فقط.

- بعد إظهار أرقام الفواتير التي تخص الاعتماد تظهر رسالة لتحث المستفيد على إدخال رقم الفاتورة التي يريدّها ويتم قبول الرقم من المستفيد (سطر ٤٣ - ٤٤) (انظر شكل ١٨ - ٦).

- نفتح ملف الفهرس PUR_INV.NDX لأنه يرتب الملف طبقاً لأرقام الفواتير (سطر ٤٥). ونبحث عن الفاتورة التي أدخلت (سطر ٤٦) فإذا وجدت يتم الآتي :

B111(a) belong to order no. 33
=====

8222
8222
A310

Enter requested invno no. to display

شكل ١٨ - ٦ إظهار أرقام فواتير الاعتماد

- * تخزين رقم الخصم بحقل ذاكرة (سطر ٤٨).
- * ربط ملف المشتريات مع ملف المخزون الرئيسي طبقاً لرقم الصنف بحيث يربط السجل الذي يحمل نفس الرقم من الملفين (سطر ٤٩).
- * يتم تصفية الملف ليقصر على السجلات التي تخص نفس الاعتماد ونفس الفاتورة فقط تمهيداً لإظهارها بعد ذلك (سطر ٥٠).
- * يتم استدعاء الإجراء BIL_FMT لإظهار محتويات الفاتورة وسبق مناقشة هذا الإجراء.
- * بعد ذلك يُسأل المستفيد عما إذا كان يريد الاستفسار عن فاتورة أخرى من فواتير الاعتماد. فإذا أجاب بلا (N) أنهيت الدوارة. وخرجنا للدوارة الخارجية (سطر ٥٢ - ٥٩).

الاستفسار عن مشتريات خلال فترة محددة

الاختيار الرابع في قائمة المشتريات للاستفسار عن مشتريات خلال فترة محددة وتحديد الفترة بإدخال تاريخ بدايتها وتاريخ نهايتها.

والبرنامج اللازم لهذا الغرض هو برنامج PURPRD.PRG ويوضح شكل ١٩ - ٦ محتويات هذا البرنامج والبرنامج لا يستدعي برامج أو إجراءات أخرى. وإنما فقط يستدعي تقرير معد سلفاً بواسطة مصمم التقارير الموجود في قاعدة البيانات لطبع التقرير الذي يوافق الفترة المطلوبة.

```

1 * -----*
2 * Program      : PURPRD.PRG                      *
3 * Author       : Magdi M. Abu Al-Ata             *
4 * Date        : October 24, 1990                  *
5 * Purpose      : Displays purchases during a period *
6 * Called from: PURMENU.PRG                        *
7 * Called from: DPAPP.PRG                          *
8 * -----*
9 sdate=space(8)
10 edate=space(8)
11 USE purch INDEX ordinv
12 @24,0
13 @24,5 SAY "Enter start date " GET sdate PICTURE "99/99/99"
14 READ
15 @24,40 SAY "Enter end date " GET edate PICTURE "99/99/99"
16 READ
17 sdate = CTOD(sdate)
18 edate = CTOD(edate)
19 dated= "date >= sdate .AND. date <= edate"
20 reptitle="Report period: "+dtoc(sdate)+" to: "+dtoc(edate)
21 @24,0
22 @24,1 SAY "Be sure your printer is ready and press ENTER key"
23 READ
24 @24,0
25 @ 24,1 SAY "***** Report printing. Please wait *****"
26 SET CONSOLE OFF
27 REPORT FORM purprt FOR &dated HEADING reptitle TO PRINT
28 SET CONSOLE ON
29 @ 24,0
30 RETURN
31
32 *****
33 * End of file PURPRD.PRG *
34 *****

```

- والبرنامج لا يشتمل على أفكار جديدة ولا أوامر معقدة مثل البرامج السابقة.
- وهو يبدأ بقبول تاريخ بداية الفترة ويخزنه في عبارة حرفية (سطر ١٣ - ١٤).
- وكذلك تاريخ نهاية الفترة (سطر ١٥ - ١٦).
- يتم تحويل العبارة الحرفية التي تشتمل على التاريخ إلى عبارة تاريخية (سطر ١٧ - ١٨).
- يتم تخزين الشرط الذي سيطبع التقرير طبقاً له داخل حقل ذاكرة (سطر ١٩)
- وهذا الشرط هو التواريخ التي تزيد عن أو تساوي تاريخ البداية والتواريخ التي تقل عن أو تساوي تاريخ النهاية.
- يتم تخزين عنوان التقرير داخل حقل ذاكرة وهو يشتمل على تاريخ البداية والنهاية (سطر ٢٠).
- قبل طباعة التقرير تظهر رسالة لتذكر المستفيد بتجهيز الطابعة فقد تكون مطفأة أو بحاجة إلى ورق. فإذا كانت الطابعة جاهزة ضغط مفتاح الإدخال لبدء الطباعة. (سطر ٢٢).
- لأن التقرير سيذهب إلى الطابعة فستظهر رسالة أمام المستفيد لتطمئنه أن التقرير يطبع حتى لا يظن أن الحاسب توقف (سطر ٢٥).
- بعد إظهار الرسالة يتم توقيف الشاشة حتى لا تستقبل التقرير مع الطابعة (سطر ٢٦).
- يبدأ التقرير في الطباعة (سطر ٢٧) وبعد الانتهاء من الطباعة تعود الشاشة إلى وضعها الطبيعي (سطر ٢٨).
- الأمر الذي يستدعي التقرير للطباعة هو أمر REPORT FORM (سطر ٢٧).
- ويجب أن يكون ملف PURPRT.FRM موجوداً على نفس الدليل وسبق تجهيزه.
- والأمر يشترط أن يطبع التقرير طبقاً للشرط المخزن بحقل dated وأن يستخدم العنوان المخزن بحقل reptitle والاختيار TO PRINT لتوجيه التقرير إلى الطابعة.
- وسبق أن شرحنا في الفصول السابقة كيفية إعداد التقرير باستخدام مصمم التقارير.

ويشتمل شكل ٢٠ - ٦ على شكل التقرير المستخرج بهذا البرنامج.

Page No. 1 Report period: 01/02/91 to: 28/02/91
03/03/91

Item Quantity Price Date
No.

** Order no. A10-5

* Bill No. S334

R-25C 50 900.00 13/02/91

S-312 10 1800.00 13/02/91

* Subsubtotal *

60 2700.00

** Subtotal **

60 2700.00

** Order no. 33

* Bill No. A310

150 33 12.00 09/02/91

77777 4 78.00 09/02/91

* Subsubtotal *

37 90.00

* Bill No. S222

150 20 1000.00 02/02/91

list8 10 765.00 02/02/91

33333 26 123.00 02/02/91

* Subsubtotal *

56 1888.00

** Subtotal **

93 1978.00

*** Total ***

153 4678.00

شكل ٢٠ - ٦ تقرير مشتريات شهر فبراير ١٩٩١م

الفصل السابع

المبيعات

Sales

يشرح هذا الفصل نظام المبيعات وكيفية تسجيل فاتورة البيع ومردودات المبيعات واستخراج تقارير البيع. والنظام يرتبط بنظام المخزون الرئيسي الذي أوردناه في الفصل الخامس ويتعامل معه. وبانتهاء دراسة هذا الفصل تكون قد انتهينا من نظام المخازن الذي بدأناه في الفصلين السابقين.

ويبدأ الفصل بشرح وظائف النظام ثم يشرح الملفات المطلوبة ثم يشرح بالتفصيل كل برنامج من برامج النظام على حدة وعلاقته بغيره من البرامج الأخرى داخل النظام.

في هذا الفصل سنشرح كيفية تسجيل المبيعات ومردوداتها والاستفسار عنها ويعتبر هذا الفصل هو الحلقة الأخيرة في نظام المخازن بصفة عامة الذي بدأناه في الفصلين السابقين.

ونظام المبيعات الذي سنشرحه هنا يقوم بالوظائف التالية:

- ١ - تسجيل المبيعات في شكل فواتير بيع ويمكن طباعة الفاتورة بعد الانتهاء من تسجيل بياناتها.
- ٢ - معالجة مردودات المبيعات أو الأخطاء التي تقع بفاتورة البيع بعد تسجيلها .
وخصم كميات المبيعات من أرصدة المخازن الرئيسية.
- ٣ - استخراج تقارير عن المبيعات وقد اكتفينا هنا بتقرير عن المبيعات خلال مدة زمنية ويمكنك الاستفادة من برامج التقارير التي شرحناها في الفصول السابقة للحصول على تقارير أخرى مثل تقارير عن مبيعات صنف معين خلال فترة معينة أو قبل أو بعد تاريخ معين . . . إلخ .

والبرامج التي سنشرحها في هذا الفصل للقيام بالوظائف السابقة برامج طويلة ومعقدة نسبياً عن تلك التي شرحناها من قبل وتشتمل على أفكار جديدة. ولذلك ننصحك بمراجعة الأنظمة السابقة إذا وجدت بعض الأفكار أو الأوامر الغامضة أو التي لم نتناولها بالشرح هنا.

تصميم وإنشاء ملفات النظام

تتطلب برامج هذا النظام ملفين رئيسيين هما:

الأول: ملف المخازن الرئيسي (INV.DBF) وقد شرحناه في الفصل الخامس .
ونحتاج إليه للتحقق من وجود رقم الصنف قبل تسجيل عملية البيع ولمعرفة البيانات المتصلة بهذا الصنف والتي ستظهر في فاتورة البيع مثل اسم الصنف وسعره بدلاً من تكرار هذه الحقول في ملف المبيعات .

الثاني: ملف المبيعات (SALE.DBF) وهو الملف التاريخي الذي سيشتمل على

المبيعات . ويتم تسجيل المبيعات بهذا الملف من واقع فواتير البيع . ونحتاج لهذا الملف لاستخراج تقارير البيع . ولتعديل أرصدة المخزون الرئيسي بعد كل فاتورة بيع بطرح الكميات المباعة من الأرصدة الموجودة بالمخازن .

ولإنشاء ملف SALE.DBF أدخل الأمر الآتي من نقطة المحث :

.CREATE SALE

ثم أدخل المواصفات الموجودة بشكل ١ - ٧ . ومنه تجد أن ملف المبيعات يشتمل على البيانات التالية :

Bytes remaining: 3952

CURSOR < — —> Char: + + Word: Home End Pan: ^+ ^+	INSERT Char: Ins Field: ^N Help: F1	DELETE Char: Del Word: ^Y Field: ^U	Up a field: ↑ Down a field: ↓ Exit/Save: ^End Abort: Esc
---	---	---	---

Field	Field Name	Type	Width	Dec		Field Name	Type	Width	Dec
1	ITEM_NO	Character	5						
2	INV_NO	Numeric	5	0					
3	QTY_SOLD	Numeric	5	0					
4	DATE_SOLD	Date	8						
5	SALESMAN	Character	12						
6	CUSTOMER	Character	12						
7	UPDATED	Logical	1						

MODIFY STRUCTURE <C> SALE 0 Field: 7/7 Caps

Enter the field name.

Field names begin with a letter and may contain letters, digits and underscores

شكل ١ - ٧ مواصفات ملف المبيعات

١ - رقم الصنف (ITEM_NO) . ويجب أن يشتمل على نفس المواصفات التي يشتمل عليها رقم الصنف بملف المخزون الرئيسي وهي نفس النوع ونفس الطول ونفس الاسم . وذلك لأن رقم الصنف سنستخدمه كمفتاح لربط الملفين داخل البرامج .

- ٢ - رقم فاتورة البيع (INV_NO) وهذه الأرقام يعطيها النظام تلقائياً مسلسلة بترتيب إصدارها.
- ٣ - الكمية المباعة (QTY_SOLD) .
- ٤ - تاريخ البيع (DATE_SOLD) .
- ٥ - اسم البائع (SALESMAN) .
- ٦ - اسم العميل (CUSTOMER) .
- ٧ - حقل منطقي ليعطي دلالة لبرنامج التعديل عما إذا كان تم تعديل رصيد هذا الصنف بعد بيعه في ملف المخازن الرئيسي أم لا (UPDATED) . وهو مشابه لهذا الحقل في ملف المشتريات الذي شرحناه في الفصل السابق .

إنشاء ملفات الفهرسة

يحتاج هذا النظام لملف فهرس يتم فيه ترتيب فواتير البيع تبعاً لأرقامها . وفهرس آخر يتم فيه ترتيب الملف تبعاً لرقم الصنف . ورغم أن أرقام الفواتير في هذا الملف ستكون مرتبة تلقائياً لأن النظام هو الذي يعطي هذه الأرقام فإننا سنحتاج لهذا الفهرس للبحث فيه عن الفاتورة التي نرغب في معالجة بياناتها بالحذف أو التعديل إذا حصل مردودات مبيعات أو حدث خطأ أثناء تسجيلها . وكما هو معروف أن أوامر البحث SEEK أو FIND تتطلب وجود ملف فهرس .

أنشئ ملفات الفهرسة المطلوبة بعد إنشاء وحفظ ملف المبيعات (SALE.DBF) هكذا :

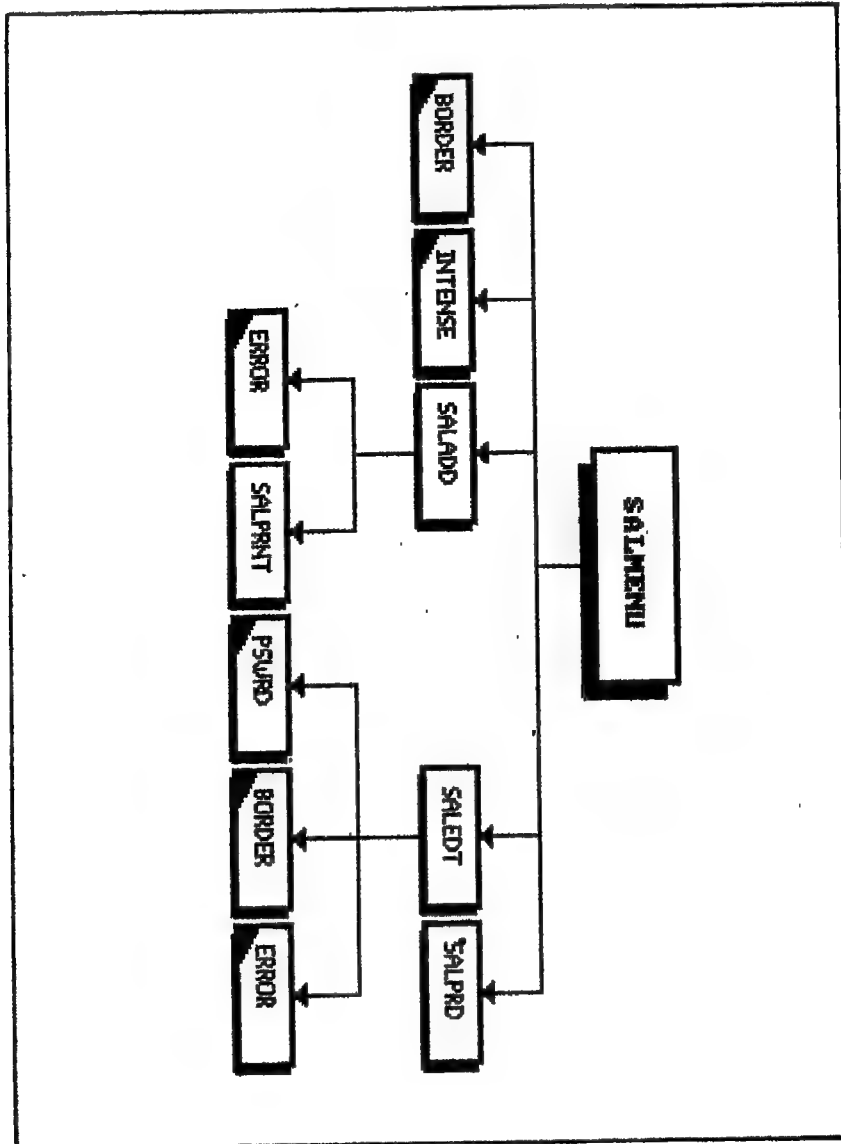
.INDEX ON INV_NO TO SALINV

.INDEX ON ITEM_NO TO ITEM_P

بهذا يصبح عندك ملف فهرس اسمه SALINV.NDX مرتباً طبقاً لأرقام الفواتير وملف آخر اسمه ITEM_P.NDX مرتباً طبقاً لأرقام الأصناف .

تحديد برامج النظام

الخطوة التالية في تطوير النظام هي رسم خريطة للنظام توضح البرامج التي يشتمل عليها. والتي تؤدي الوظائف المطلوبة من النظام. ويوضح شكل ٢ - ٧ خريطة النظام التي توضح العلاقة بين البرامج التي يشتمل عليها. ومنه يتضح أن



شكل ٧ - ٢ خريطة نظام المبيعات

البرنامج الرئيسي وهو SALMENU.PRG يستدعي ثلاثة برامج أخرى كل منها يقوم بوظيفة محددة ويستدعي أحدها وهو SALADD برنامجاً تابعاً وهو SALPRNT . ويمثل كل وظيفة من هذه الوظائف مستطيل داخل خريطة النظام بالإضافة إلى مجموعة إجراءات تعرفها بتظليل الركن اليسار السفلي من المستطيل . وكل هذه الإجراءات مألوفة لنا من شرح الفصول السابقة باستثناء إجراء واحد فقط جديد وهو psword . وستناول فيما يلي شرح البرامج الموجودة بالخريطة بالإضافة إلى الإجراء الجديد .

قائمة نظام المبيعات

يشتمل شكل ٣ - ٧ على قائمة نظام المبيعات التي تنفرع عن القائمة الرئيسية . وكما تلاحظ فإنها تشتمل على الوظائف الرئيسية المطلوبة من النظام وهي تسجيل فواتير البيع ومعالجة مردودات المبيعات وتقارير المبيعات . كما يشتمل شكل

Tuesday 26, February 1991
Time: 15:26:37

Arabian Computer Center
C.R. 39387

Sales menu

- (1) Add sales.....
- (2) Modify sales...
- (3) Sales during a Period
- (6) Go to main menu
- (Q) Quit to DOS....

Select number or colored letter ==> █

شكل ٣ - ٧ قائمة نظام المبيعات

٤ - ٧ على برنامج SALMENU.PRG اللازم لإظهار هذه القائمة وتنفيذ اختياراتها ولا تختلف فكرة هذا البرنامج عن غيره من برامج القوائم التي شرحناها سابقاً.

```

1 * -----*
2 * Program.....: SALMENU.PRG*
3 * Called form.: DBAPP.PRG*
4 * Programmer...: Magdi M. Abu Al-Ata*
5 * Date.....: Dec. 25,1990*
6 * Purpose.....: To display sales menu*
7 * -----*
8 PRIVATE scrtitle,scrfootr,choice
9 scrtitle = "Sales menu"
10 scrfootr = "Select number or colored letter ==>"
11 ok = .T.
12 DO WHILE ok
13     choice = " "
14     DO border WITH scrtitle,scrfootr
15     * Draw menu
16     @10, 5 SAY "(1) Add sales....."
17     DO intense WITH 10,10,"A"
18     @11, 5 SAY "(2) Modify sales..."
19     DO intense WITH 11,10,"M"
20     @12, 5 SAY "(3) Sales during a Period"
21     DO intense WITH 12, 25,"p"
22     @13, 5 SAY "(G) Go to main menu"
23     DO intense WITH 13,10,"G"
24     @14, 5 SAY "(Q) Quit to DOS...."
25     DO intense WITH 14,10,"Q"
26     @22,43 GET choice
27     READ
28     * Do selected option
29     DO CASE
30     CASE choice $ "1aA "
31         DO saladd                                && Add new invoice
32     CASE choice $ "2mM"
33         DO saledt                                && Edit an invoice
34     CASE choice $ "3pp"
35         DO salprd                                && Display sales
36     CASE UPPER(choice) = "G"
37         RETURN                                    && Go to Main menu
38     CASE UPPER(choice) = "Q"
39         CLEAR
40         QUIT                                    && Quit to DOS
41     OTHERWISE
42     LOOP
43     ENDCASE
44 ENDD
45 *****
46 * End of file SALMENU.PRG *
47 *****

```

شكل ٤ - ٧ برنامج SALMENU.PRG

تسجيل فواتير البيع

بمجرد اختيار الرقم ١ أو الحرف A من قائمة المبيعات (شكل ٣ - ٧ السابق) يقوم البرنامج بإعطاء رقم مسلسل للفاتورة تلقائياً ويظهر شاشة تشتمل على رقم الفاتورة بالإضافة إلى بيانات الشركة ويطلب اسم البائع واسم العميل. ويشتمل شكل ٥ - ٧ على الشكل المبدي للفاتورة قبل تعبئة بياناتها. فإذا أدخل المستخدم اسم

Arabian Computer Center C.R. 39387 Cash Invoice				
Sale date: Monday 25 February 1991 Salesman:			Invoice No. 13 Sold to:	
Item No.	Description	Quantity	Price	Amount

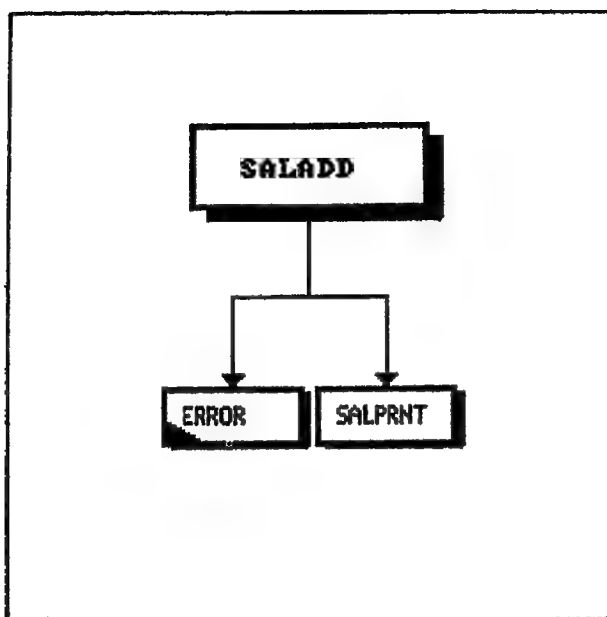
شكل ٥ - ٧ الشكل المبدي لفاتورة البيع

البائع واسم العميل يطلب البرنامج إدخال بيانات الأصناف المباعة ويقوم بتجميع الفاتورة وحساب الخصم إن وجد وإعطاء المجموع النهائي. ويتم طباعة الفاتورة على الطابعة إذا أراد المستخدم ذلك ويشتمل شكل ٦ - ٧ على الشكل النهائي لفاتورة بيع تشتمل على أربعة أصناف.

والبرنامج الذي يقوم بهذه المهمة هو SALADD.PRG ويشتمل شكل ٧ - ٧ على الخريطة التي توضح علاقة هذا البرنامج بغيره من البرامج داخل النظام. ومنها

Arabian Computer Center C.R. 39387 Cash Invoice				
Sale date: Monday 25, February 1991			Invoice No. 13	
Salesman: Mansour			Sold to: Ebad Salem	
Item No.	Description	Quantity	Price	Amount
8-324	Hard Disk	5	1,100.00	5,500.00
8-445	Modem	2	500.00	1,000.00
8-312	Printer	1	2,000.00	2,000.00
8-319	Monitor	2	350.00	700.00
Total				9,200.00
Discount				200.00
Final Total				9,000.00
=====				
Do you want to print this invoice? [Y/N] █				

شكل ٦ - ٧ الشكل النهائي لفاتورة البيع



شكل ٧ - ٧ خريطة برنامج SALADD.PRГ

يتضح أن البرنامج يستدعي الإجراء error لإظهار رسالة الخطأ وبرنامج
SALPRNT.PRG لطباعة الفاتورة على الطابعة. كما يشتمل شكل ٨ - ٧ على برنامج
SALADD.PRG ونظراً لأهمية هذا البرنامج فإننا سنشرح أولاً فكرة البرنامج وطريقة
عمله ثم نشرح خطوات البرنامج والأوامر الجديدة التي يشتمل عليها. وبعد ذلك
نشرح برنامج طباعة الفاتورة الذي يستدعيه.

```

1 * -----*
2 * Program      : SALADD.PRG                      *
3 * Author       : Magdi M. Abu Al-Ata            *
4 * Date         : October 24, 1990                *
5 * Purpose      : Records transaction sales in SALE.DBF *
6 * Called from: SALMENU.PRG                      *
7 * Called from: DPAPP.PRG                        *
8 * -----*
9 PRIVATE msg,more,minv
10 msg = " Press "+CHR(17)+CHR(196)+CHR(217)+" to continue"
11 more = " "
12 * Get last used invoice No. from SALE.DBF
13 USE sale
14 GO BOTTOM
15 minv = inv_no
16 * Add new transaction to temporary file
17 SET SAFETY OFF          && Disables the warning message when the
18                          && file is about to be overwritten.
19 COPY STRUCTURE TO tmpsal
20 SET SAFETY ON           && Let the warning message to appear
21 * Open master and temporary files
22 SELECT A
23 USE inv
24 SELECT B
25 USE tmpsal
26 * Set up a loop to display invoice form.
27 DO WHILE .T.
28     minv = minv+1
29     STORE SPACE(12) TO msalman,mcust
30     mtotat = 0
31     * Write Invoice title on screen
32     CLEAR
33     @ 0,80/2-(LEN(systitle)/2) SAY systitle
34     @ 1,80/2-(LEN(title2)/2) SAY title2
35     @ 2,80/2-(LEN("Cash Invoice")/2) SAY "Cash Invoice"
36     @ 3, 2 SAY "Sale date: "+ mdate
37     @ 3,50 SAY "Invoice No. " + STR(minv,4)
38     @ 4, 2 SAY "Salesman: " GET msalman
39     @ 4,50 SAY "Sold to: " GET mcust
40     @ 5, 0 SAY REPLICATE(CHR(45),79)
41     ?" Item No.  Discription ",SPACE(13)

```

شكل ٨ - ٧ برنامج SALADD.PRG

```

42      ??"Quantity  Price      Amount "
43      @ 07,0 SAY REPLICATE(CHR(45),79)
44      READ
45      * Set up a loop for each item on the invoice
46      row = 08
47      plus = .T.
48      DO WHILE plus
49          * Set up invoice memory variables
50          mqty = 0
51          itemno = SPACE(5)
52          @row,2 GET itemno PICTURE "@1"
53          READ
54          * Make sure item no. exists
55          itemno = TRIM(itemno)
56          SELECT inv
57          SET INDEX TO item              && Indexed on item_no
58          SEEK itemno
59          * Decide next step based on existence of item no.
60          DO CASE
61              CASE LEN(itemno) = 0      && No item no. entered
62                  <-----EXIT
63                  CASE .NOT. FOUND()    && Item not exist
64                      DO error WITH "This item not found. "
65                      row = row - 1      && Because we always add 1 to
66                                      && line no. for each item
67                  CASE FOUND()          && Item no. is exist
68                      * B sure the item not included in this invoice
69                      SELECT tmpsal
70                      SET FILTER TO inv_no = minv
71                      GO TOP
72                      LOCATE FOR item_no = itemno
73                      IF FOUND()
74                          DO error WITH "This item has been selected "
75                  -----LOOP
76                  ENDIF
77                  SET FILTER TO
78                  SELECT inv
79                  @row,15 SAY desc      && Say item name
80                  mprice = price
81                  @row,40 GET mqty PICTURE "99,999"
82                  READ
83                  * If quantity = 0 or exceeds the quantity on
84                  * stock give a message and loop. Else display total
85                  IF mqty = 0 .OR. mqty > onstock
86                      DO error WITH ;
87                      " Quantity in stock is " + ltrim(str(onstock))
88                      @ row,10          && Erase data entered to this line
89                  -----LOOP
90                  ELSE
91                      * IF requested quantity put the inventory in
92                      * reorder point give a warning message
93                      valid = onstock - reorder
94                      IF mqty >= valid
95                          DO error WITH " Quantity on stock "+ ;

```

```

        ltrim(str(onstock))+" reorder point is "+ ;
        ltrim(str(reorder))
    ENDIF
    @row,48 SAY mprice PICTURE "99,999.99"
    @row,60 SAY mqty * mprice PICTURE "9,999,999.99"
    mtotat = mtotat + (mqty * mprice)
ENDIF
* Add a blank record to the tmpsal file,
* and fill its fields
SELECT tmpsal
APPE BLANK
REPLACE date_sold WITH DATE(), salesman WITH msalman, ;
customer WITH mcust, inv_no WITH minv, ;
item_no WITH itemno, qty_sold WITH mqty
ENDCASE
row = row+1
* Scroll screen if nearing end
IF row >= 19
    @24,1
    ?
    row = 19
ENDIF
ENDDO (While still adding transactions to invoice)
* Display total and pause before next invoice
@ row+1,42 SAY "Total "
@ row+1,60 SAY mtotat PICTURE "999,999.99"
* Calculate discount
redu = 0.00    && Initialize memvar for discount
@ row+2,42 SAY "Discount "
@ row+2,62 GET redu PICTURE "9,999.99"
READ
ftotal = mtotat - redu    && Put final total in memvar
@ row+3,60 SAY "-----"
@ row+4,42 SAY "Final Total "
@ row+4,60 SAY ftotal PICTURE "999,999.99"
@ row+5,60 SAY "-----"
pinv = "?"
@ 24,2 SAY " Do you want to print this invoice? [Y/N] " ;
GET pinv PICTURE "!"
READ
IF pinv = "Y"
    DO salprnt    && Print invoice, reset transactions
ENDIF
@ 24,2
@ 24,2 SAY " Do another one? [Y/N] " GET more PICTURE "!"
READ
IF more = "N"
<-----EXIT
ENDIF
ENDDO ( Add invoice while user doesn't request exit )
CLOSE DATABASES
USE sale
APPEND FROM tmpsal
ERASE tmpsal.dbf

```

```
144 CLOSE DATABASES      && close database and its associated index
145 * Update inventory file from sales file
146 * DO updtal
147 RETURN
148
149 *****
150 *   End of file SALADD.PRG *
151 *****
```

تابع شكل ٨ - ٧ برنامج SALADD.PRG

يشتمل البرنامج على دوارتين متداخلتين. تتكرر الكبرى (الخارجية) طالما أن المستفيد يرغب في إصدار فاتورة أخرى. وتتكرر الصغرى (الداخلية) طالما أن المستفيد يرغب في إضافة أصناف جديدة إلى الفاتورة وفي هذه الدوارة الداخلية يتأكد البرنامج أولاً من وجود رقم الصنف المُدخَّل في ملف المخازن الرئيسي ويتأكد بعد ذلك أن هذا الرقم غير مسجل بنفس فاتورة البيع قبل ذلك فإذا تحقق من ذلك يظهر اسم الصنف تلقائياً من ملف المخزون الرئيسي ويطلب إدخال الكمية المباعة. فإذا أدخلت الكمية يتأكد أولاً أنها تساوي أو تزيد عن الكمية الموجودة بملف المخزون لأننا لا نستطيع بيع كمية أكثر من الموجودة عندنا.

فإذا تأكد من ذلك يتأكد أن الكمية التي ستبقى بالمخازن بعد البيع لن تصل إلى حد إعادة الطلب. فإذا كانت ستصل إلى حد إعادة الطلب يظهر رسالة تحذيرية فقط تخبر عن نقطة إعادة الطلب والكمية المتبقية. بعد ذلك يظهر السعر تلقائياً. وفي النهاية يحسب المجموع والخصم إن وجد والمبلغ النهائي. ويقوم بطباعة الفاتورة حسب طلب المستفيد.

والآن نعود لشرح خطوات البرنامج وطريقة عمله:

- يبدأ البرنامج بتخصيص حقول ذاكرة لتستخدم داخل هذا البرنامج فقط. (سطر رقم ٩).

- ينشئ حقول ذاكرة لتستخدم داخل هذا البرنامج فقط (سطر ١٠ ، ١١) وقد أنشأنا حقل باسم msg وكما تذكر فإن هذا الحقل بهذا الاسم موجود في ملف

- الإجراءات DBPROC.PRG الذي تستدعيه كل برامج النظام . ولكننا قصدنا هنا تغيير هذه الرسالة في هذا البرنامج لتعطي عبارة to continue بدلاً من عبارة to try again لأن الرسالة التي تظهر إذا وصلت كمية المخزون بعد البيع إلى حد إعادة الطلب تطلب الاستمرار في العمل وليس إعادة المحاولة كما يظهر دائماً عند الخطأ .
- يذهب البرنامج إلى آخر سجل بملف المبيعات ليعرف رقم آخر فاتورة ويضعه في حقل ذاكرة (سطر ١٣ - ١٥) .
- ننسخ مواصفات ملف البيع إلى ملف جديد اسمه DBF.TMPSAL لأننا سنستخدم هذا الملف لنضع فيه الأصناف المباعة طوال فترة تشغيل البرنامج . وفي نهاية اليوم أو عند إغلاق البرنامج يتم نقل محتوياته إلى ملف البيع وحذفه نهائياً . والغاية من ذلك أن يتم تسجيل المبيعات بعيداً عن الملف الحقيقي حتى لا يتأثر ملف المبيعات إذا ألغيت الفاتورة بعد كتابتها وأن يتم طباعة الفاتورة من هذا الملف لأنه سيكون ملف مؤقت يشتمل على آخر فاتورة أو آخر فواتير تم تسجيلها بعد فتح البرنامج . (سطر ١٩) .
- حتى لا تظهر للمستفيد في المرة التالية لتشغيل البرنامج رسالة تفيد وجود الملف المؤقت قبل الكتابة عليه وضعنا أمر SET SAFETY في حالة OFF (سطر ١٧) ثم أعدناه إلى حالته الطبيعية (سطر ٢٠) .
- نفتح ملف المخازن وملف المبيعات المؤقت معاً (سطر ٢٢ - ٢٥) وقد أوضحنا الهدف من الملف المؤقت أما ملف المخازن (INV.DBF) فإننا نحتاج إليه لمعرفة ما إذا كان الصنف موجوداً فعلاً بالمخازن أم لا ولمعرفة اسمه وسعره بعد ذلك ولمقارنة الكمية المطلوبة بالكمية الموجودة لتحديد هل الكمية مقبولة أم لا .
- تنشأ دوائر تستمر طالما أن المستفيد لم يطلب إنهاء البرنامج وطالما يريد إصدار فواتير جديدة (سطر ٢٧ - ١٣٩) وفي داخل هذه الدوائر يتم الآتي :
 - إضافة واحد لرقم آخر فاتورة موجود بملف البيع (سطر ٢٨) .
 - إنشاء حقول بالذاكرة لتستقبل اسم البائع واسم العميل ومجموع الفاتورة (سطر ٢٩ - ٣٠) .
 - يتم كتابة اسم الشركة وعنوانها والتاريخ وتسجيل الفاتورة وطلب إدخال اسم البائع

- واسم العميل (سطر ٣٢ - ٤٤) ويتم كتابة اسم وعنوان الشركة بنفس الطريقة التي تظهر بها في شاشات النظام (راجع الفصل الثالث).
- يحتل عنوان الشركة وتسطير الفاتورة ٧ أسطر من الشاشة ويجب أن يتم تسجيل أصناف الفاتورة بعد ذلك ولذلك وضعنا الرقم ٨ في حقل ذاكرة اسمه row لاستخدامه في البرنامج لتوضيح مكان إدخال أول صنف.
- سننشأ دارة داخل الدارة الأولى تستمر طالما أن المستفيد يكتب اسم صنف جديد (سطر ٤٨ - ١١٢) وفي داخل هذه الدارة يتم الآتي.
- إنشاء حقول ذاكرة لتستقبل الكمية ورقم الصنف (سطر ٥٠ - ٥١).
- يتم قبول رقم الصنف وحذف الفراغات إن وجدت على يمينه (سطر ٥٢ - ٥٥).
- نختار ملف المخزون (INV.DBF) ليكون هو الملف الحالي مرتباً تبعاً لأرقام الأصناف ثم نبحث عن الرقم الذي أدخل (سطر ٥٦ - ٥٨).
- يتم تنفيذ واحدة من ثلاث حالات:
- الأولى: إدخال فراغات محل رقم الصنف مما يسبب الخروج من الدارة وإنهاء تسجيل الأصناف (سطر ٦١ - ٦٢).
- الثانية: أن تظهر رسالة تفيد عدم وجود الصنف بالمخازن وبعدها نطرح واحد من قيمة row (سطر ٦٣ - ٦٥) لأننا نضيف إليها باستمرار واحد لننتقل إلى السطر التالي (انظر سطر ١٠٦).
- الثالثة: إذا كان رقم الصنف موجود بالمخازن نختار الملف المؤقت ليكون هو الحالي (سطر ٦٩) ونخصص به فقط السجلات التي تشتمل على نفس رقم الفاتورة (سطر ٧٠) ثم نبحث فيه عن رقم الصنف (سطر ٧٢) فإذا كان موجوداً (سطر ٧٣) فهذا يعني أن هذا الصنف سبق تسجيله بنفس الفاتورة ولذلك تظهر رسالة للمستفيد تعلمه أن هذا الصنف اختير بنفس الفاتورة قبل ذلك (سطر ٧٤) لأنه لا يصح أن تباع صنف واحد مرتين في فاتورة واحدة. ثم نعاود المحاولة لكتابة صنف آخر (سطر ٧٥). ثم نعيد الملف إلى حالته الأصلية (سطر ٧٧).
- بعد اختيار رقم الصنف نعود لاختيار ملف المخزون ليكون هو الملف الحالي (سطر

- ٧٨) ثم نكتب منه اسم الصنف (سطر ٧٩). ونضع السعر المسجل داخل حقل ذاكرة (سطر ٨٠).
- يُطلب إدخال الكمية (سطر ٨١ - ٨٢) وبعد إدخال الكمية تتم مراجعتها فإذا كانت تساوي صفر أو تزيد عن الموجود بالمخازن تظهر رسالة خطأ ونعود لبداية الدوارة لإدخال الصنف مرة ثانية (سطر ٨٥ - ٨٨).
- أما إذا كانت الكمية تزيد عن صفر وتقل عن الموجود بالمخازن فيتم خصمها من الموجود بالمخازن لمعرفة هل المتبقى من هذا الصنف وصل إلى حد إعادة الطلب أم لا. (سطر ٨٩ - ٩٣).
- إذا كان المتبقى وصل إلى حد إعادة الطلب تظهر رسالة تحذيرية فقط وبها كمية المخزون وحد إعادة الطلب ليفهم المستفيد أن هذا الصنف وصل إلى الضوء الأصفر إلا أن عملية البيع تستمر ولا تتوقف لأن المنطقي أن نبيع ما دمنا نملك الكمية (سطر ٩٤).
- إذا قبلت الكمية ظهر السعر تلقائياً والمبلغ المستحق نتيجة ضرب الكمية \times السعر (سطر ٩٦ - ٩٧).
- يتم إضافة حاصل الضرب في الخطوة السابقة إلى المجموع (mtotal) (سطر ٩٨).
- حتى الآن تم تسجيل بيانات الصنف المباع بحقول داخل الذاكرة ويجب نقلها إلى الملف المؤقت الذي نستخدمه ولذلك يتم اختيار الملف (سطر ١٠٢) وإضافة سجل خال في نهاية (سطر ١٠٣) ثم استبدال حقوله بالبيانات المناظرة من حقول الذاكرة (سطر ١٠٤).
- إلى هنا تنتهي الدوارة الداخلية التي تتكرر طالما ندخل صنفاً جديداً.
- قبل تسجيل صنف جديد ننتقل للسطر التالي (سطر ١٠٦).
- السطورية من ١٠٨ إلى ١١٢ تنطوي على فكرة تغلب على مشكلة ستواجهنا إذا انتهت سطور الشاشة ولم تنته الأصناف التي نريد تسجيلها بالفاتورة وهي طي الشاشة سطر واحد لأعلى لقبول سطر جديد.
- بعد الانتهاء من تسجيل الأصناف المباعة بالفاتورة يظهر المجموع بعد سطر من آخر صنف (سطر ١١٥ - ١١٦).

- يطلب البرنامج إدخال الخصم إن وجد (سطر ١١٨ - ١٢١) ثم يحسب الصافي النهائي للفاتورة بطرح مبلغ الخصم من المجموع ووضع الناتج في حقل ذاكرة (سطر ١٢٢).
- يتم كتابة المجموع النهائي بين خطين في الموقع المناسب على الشاشة (سطر ١٢٣ - ١٢٦).
- إلى هنا انتهت الفاتورة وستظهر رسالة تسألك هل تريد طباعتها فإذا كانت الإجابة نعم يتم استدعاء برنامج SALPRNT.PRG لطباعتها. سنناقش هذا البرنامج بعد قليل. (سطر ١٢٧ - ١٣٢).
- ثمحي الرسالة الموجودة بسطر ٢٤ لتحل محلها رسالة جديدة (سطر ١٣٣).
- سواء اخترت طباعة الفاتورة أم لا سيسألك البرنامج هل تريد الاستمرار في بيع فاتورة أخرى أم الخروج من البرنامج ويتقرر تكرار بيع فاتورة أم لا بناء على رغبتك (سطر ١٣٤ - ١٣٨).
- قبل الخروج من البرنامج لابد من تنفيذ الخطوة الأولى أو الخطوتين التاليتين بعد إغلاق الملفات:
الخطوة الأولى : نقل السجلات الموجودة بملف المبيعات المؤقت (TMPSAL.DBF) إلى نهاية ملف المبيعات التاريخي (SALE.DBF) (سطر ١٤١ - ١٤٢). ولأن الملف المؤقت ينتهي الغرض منه بنقل محتوياته فنحذفه لتوفير المساحة التي يشغلها (سطر ١٤٣).
- الخطوة الثانية: تعديل أرصدة الأصناف المباعة بملف المخزون الرئيسي وذلك بطرح الكميات المباعة من تلك الموجودة بالمخازن. وهذه يمكن أن تتم في هذا المكان أو باستخدام أحد برامج صيانة الملفات. وقد فضلنا الطريقة الثانية. ولذلك سنرجى شرح برنامج التعديل إلى الفصل التاسع ضمن برامج صيانة الملفات ولذلك فقد وضعت علامة * أمام الأمر الذي يستدعي برنامج التعديل في سطر ١٤٦ بدلاً من حذف هذا السطر لأذكرك أنه يمكنك هنا أو في هذا المكان إجراء التحديث المطلوب للملفات. وتقرير اختيار واحدة من الطريقتين يرجع إلى سياسة الشركة.

طباعة فاتورة البيع

قلنا أن البرنامج السابق يستدعي برنامجاً آخر ليقوم بطباعة الفاتورة حسب رغبة المستخدم والبرنامج الذي يتولى طباعة الفاتورة هو SALPRNT.PRG والبرنامج لا يستدعي برامج أو إجراءات أخرى.

ويشتمل شكل ٩ - ٧ على نص البرنامج اللازم لطباعة الفاتورة كما يشتمل شكل ١٠ - ٧ على شكل الفاتورة المطبوعة باستخدام هذا البرنامج ورغم أنه عملياً

```

1 * ----- *
2 * Program.....: salprnt.prg *
3 * Called form.: saladd.prg *
4 * Called form.: salmenu.prg *
5 * Called form.: dbapp.prg *
6 * Programmer...: Magdi M. Abu Al-Ata *
7 * Date.....: Dec. 25,1990 *
8 * Purpose.....: To print sale invoice *
9 * ----- *
10 SET DEVICE TO PRINT
11 SET CONSOLE OFF
12 SET HEAD OFF
13 *
14 SELECT INV
15 SET INDEX TO item && Indexed on item no
16 SELECT tmpsal
17 SET FILTER TO inv_no = minv
18 GO TOP
19 SET RELATION TO item_no INTO A
20 *
21 @ 1,80/2-(LEN(systitle)/2) SAY systitle
22 @ 2,80/2-(LEN(title2)/2) SAY title2
23 @ 3,80/2-(LEN("Cash Invoice")/2) SAY "Cash Invoice"
24 @ 4, 2 SAY "Sale date: " + mdate
25 @ 4,50 SAY "Invoice No. " + STR(minv,4)
26 @ 5, 2 SAY "Salesman: " + msalman
27 @ 5,50 SAY "Sold to: " + mcust
28 @ 6, 0 SAY REPLICATE(CHR(45),79)
29 @ 7,0 SAY " Item No. Description"+SPACE(9)+"Quantity"+SPACE(6)+ ;
29 + "Price"+SPACE(18)+"Amount"
30 @ 8,0 SAY REPLICATE(CHR(45),79)
31
32 * List invoice items
33 LIST OFF item_no,SPACE(3),A->desc,SPACE(5),qty_sold,SPACE(7), ;
33 A->price,SPACE(12),qty_sold * A->price TO PRINT
34

```

شكل ٩ - ٧ برنامج SALPRNT.PRG

```

35 * Display total
36 @ PROW()+3,42 SAY "Total "
37 @ PROW(),68 SAY mtotal PICTURE "999,999.99"
38 @ PROW()+2,42 SAY "Discount "
39 @ PROW(),68 SAY redu PICTURE "999,999.99"
40 @ PROW()+1,68 SAY "-----"
41 @ PROW()+1,42 SAY "Final Total "
42 @ PROW(),68 SAY ftotat PICTURE "999,999.99"
43 @ PROW()+1,68 SAY "===== "
44 EJECT
45 *
46 SET CONSOLE ON
47 SET DEVICE TO SCREEN
48 SET HEAD ON
49 SET FILTER TO
50 SET RELATION TO
51 RETURN
52 *****
53 * End of file SALPRNT.PRГ *
54 *****

```

تابع شكل ٩ - ٧ برنامج SALPRNT.PRГ

Arabian Computer Center				
C.R. 39387				
Cash Invoice				
Sale date: Tuesday 26, February 1991			Invoice No. 14	
Salesman: A. Naser			Sold to: E. Salem	
Item No.	Descreeption	Quantity	Pprice	Amount
S-319	Monitor	3	350.00	1050.00
R-445	Modem	4	500.00	2000.00
S-312	Printer	2	2000.00	4000.00
Total				7,050.00
Discount				50.00
Final Total				7,000.00

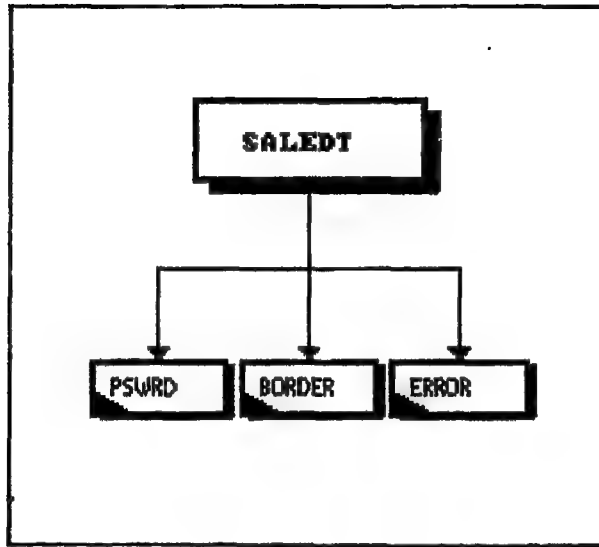
شكل ١٠ - ٧ شكل الفاتورة المطبوعة

- لا يتم طباعة الفواتير بهذا الشكل وإنما الذي يحدث أن الشركات تطبع فواتير خاصة لهذا الغرض ويقوم البرنامج بتعبئة خانات الفاتورة بالبضاعة المباعة إلا أنك يمكنك الاستفادة من أفكار البرنامج وطريقة عمله لتعبئة خانات مثل هذه الفواتير وذلك بتغيير أماكن طباعة الحقول فقط باستخدام أمر SAY...@ كما أنه بإمكانك الاستفادة من البرنامج لطباعة تقارير مماثلة وبالرجوع إلى شكل ٩ - ٧ نوضح ما يلي:
- يبدأ البرنامج بتوجيه المخرجات إلى الطابعة بدلاً من الشاشة ويتوقف عمل الشاشة أثناء طباعة الفاتورة (سطر ١٠ - ١١).
 - أمر SET HEAD OFF يلغي أسماء الحقول التي تظهر مع أمر LIST أو DISPLAY (سطر ١٢).
 - قبل طباعة الفاتورة يتم تجهيز الملفات كما يلي:
 - * نختار ترتيب ملف المخازن (INV.DBF) تبعاً لرقم الصنف (سطر ١٤ - ١٥).
 - * نختار السجلات التي تشتمل فقط على رقم الفاتورة المراد طباعتها (سطر ١٦ - ١٧).
 - * يتم ربط الملفين معاً طبقاً لرقم الصنف المشترك بينهما (سطر ١٩).
 - نطبع اسم الشركة وتسطير الفاتورة بطريقة مشابهة للفاتورة التي تظهر على الشاشة (سطر ٢١ - ٣٠).
 - نطبع أصناف الفاتورة باستخدام أمر LIST (سطر ٣٣ ويمكنك استخدام @...DAY داخل دارة لهذا الغرض).
 - يتم طباعة إجماليات الفاتورة والخصم في الأماكن المناسبة على الورق (سطر ٣٦ - ٤٣).
 - تقفز الطابعة لبداية صفحة جديدة ليتمكن المستفيد من نزع الفاتورة وإعطائها العميل. (سطر ٤٤).
 - قبل العودة للبرنامج السابق يتم إعادة تخصيص وحدات الإخراج والملفات إلى حالتها السابقة قبل بدء الطباعة (سطر ٤٦ - ٥٠).

معالجة مردودات المبيعات

الاختيار الثاني في قائمة المبيعات هو تعديل بيانات البيع (راجع شكل ٣ - ٧). ونقصد بتعديل بيانات البيع معالجة الأخطاء التي تقع في فاتورة البيع ويتم اكتشافها بعد ذلك أو معالجة مردودات المبيعات بعد أن يتم إصدار الفاتورة وتعديل أرصدة المخازن. ونظراً لحساسية هذا الأمر وحتى لا يتخذ ذريعة للتلاعب في الملفات. فقد قصرنا هذه الإمكانية على شخص مسئول لا يمكنه إجراء هذه التعديلات إلا بعد إدخال كلمة سر مناسبة.

والبرنامج الذي يتولى هذه المهمة هو برنامج SALEDT.PRG ويشتمل شكل ١١ - ٧ على الخريطة التي توضح علاقة هذا البرنامج بغيره من البرامج داخل النظام. ومنها يتضح أن البرنامج يستدعي إجراءً جديداً يظهر لأول مرة وهو pswrd سنناقشه بعد الانتهاء من مناقشة البرنامج. كما يشتمل شكل ١٢ - ٧ على محتويات البرنامج. وسنناقش أولاً فكرة البرنامج وطريقة عمله قبل مناقشة خطوات البرنامج والأوامر الجديدة التي يشتمل عليها.



شكل ١١ - ٧ خريطة برنامج SALEDT.PRG

```

1 * -----*
2 * Program.....: SALEDT.PRG*
3 * Called form.: SALMENU.PRG*
4 * Called form.: DBAPP.PRG*
5 * Programmer...: Magdi M. Abu Al-Ata*
6 * Date.....: Dec. 25,1990*
7 * Purpose.....: To edit sale invoice in case of return or error*
8 * -----*
9 PRIVATE scrttitle,scrfootr,scrtitl2,scrfoot2,minv,recno,option
10 scrttitle = "Modifying sale invoice"
11 scrfootr = "Enter invoice no. or press " + ;
11 CHR(17) + CHR(196) + CHR(217) + " to exit."
12 scrtitl2 = "Selling invoice number: "
13 scrfoot2 = "Enter [D] to Delete, [M] to Modify "
14 keepon = .t.
15 DO pswrd
16 SELECT A
17 USE sale
18 SELECT B
19 USE inv INDEX item
20 DO WHILE keepon
21     minv = 0
22     DO border WITH scrttitle,scrfootr
23     @ 16,5 SAY " Invoice No. " GET minv
24     READ
25     * if user enter 0 exit, exit the program
26     IF minv=0
27         <-----EXIT
28     ELSE
29         * Edit the invoice
30         IF minv > 0
31             SELECT sale
32             SET RELATION TO item_no INTO inv
33             * Check if the invoice found.
34             COUNT FOR inv_no = minv TO howmany
35             DO CASE
36                 * IF invoice not found give a message
37                 CASE howmany = 0
38                     DO error WITH "Invoice no." + ;
38                     LTRIM(STR(minv)) + " not found."
39                 CASE howmany > 0
40                     * If invoice has records list them to correct
41                     moredt = .T.
42                     DO WHILE moredt
43                         CLEAR
44                         @ 0,5 SAY "Invoice No. "+LTRIM(STR(minv))
45                         @ 1,5 SAY "===== " + ;
45                         REPLICATE(CHR(61),LEN(LTRIM(STR(minv))))
46                         @ 3,0 SAY REPLICATE(CHR(196),80)
47                         @ 4,0 SAY " Rec No.      Item"+SPACE(7)+ ;
47                         "Quantity"+ SPACE(6)+"Salesman"+SPACE(8)+ ;
47                         "Customer"+SPACE(10)+"Date"
48                         @5,0 SAY REPLICATE(CHR(196),80)
49                         SET HEAD OFF

```

```

50 LIST SPACE(2),item_no,SPACE(4),qty_sold, ;
50 SPACE(7),salesman,SPACE(2), customer, ;
50 SPACE(04),date_sold FOR inv_no = minv
51 SET HEAD ON
52 ?                                && Display blank record
53 INPUT " Enter record no. need to edit " TO recno
54 GO recno
55
56 * Put fields to edit in memory variables
57 old itm = item_no
58 old qty = qty_sold
59
60 * Display record data on screen
61 do border with scrtitl2+LTRIM(STR(minv)),scrfoot2
62 @ 10, 6 SAY "Item No."
63 @ 10, 28 GET item_no
64 @ 12, 6 SAY "Description"
65 @ 12, 28 GET B->desc
66 @ 14, 6 SAY "Salesman"
67 @ 14, 28 GET salesman
68 @ 16, 6 SAY "Customer"
69 @ 16, 28 GET customer
70 @ 18, 6 SAY "Quantity sold"
71 @ 18, 28 GET qty_sold PICTURE "99999"
72 @ 20, 6 SAY "Unit price"
73 @ 20, 28 GET B->price PICTURE "9,999.99"
74 @ 20, 44 SAY "Selling date"
75 @ 20, 60 GET date_sold
76 CLEAR GETS
77 option = " "
78 @ 22,43 GET option PICT "!"
79 READ
80 IF option $ "MD"
81     IF option = "M"
82         * Display record to edit
83         @ 10,4 CLEAR TO 20,75
84         @ 14, 6 SAY "Item No."
85         @ 14, 28 GET item_no
86         @ 16, 6 SAY "Quantity sold"
87         @ 16, 28 GET qty_sold PICTURE "99999"
88         READ
89     ELSE
90         yn = " "
91         @ 22, 5 SAY ;
91         "Confirm deleting? =====> [Y/N] "
92         @ 22,43 GET yn PICTURE "!"
93         READ
94         IF yn = "Y"
95             DELETE
96             * PACK
97         ENDIF (yn = "Y")
98     ENDIF (option = M)
99 ENDIF (option = M or D)
100

```

```

101      * After deleting or modifying adjust INV.DBF
102      DO CASE
103      * If record deleted and has been updated,
104      * add its quantity to INV.DBF
105      CASE UPDATED .AND. option = "D"
106      SELECT inv
107      SEEK old_itm
108      IF FOUND()
109      | REPLACE onstock WITH onstock + old_qty
110      ENDIF
111      * If record changed and has been updated,
112      * add its qty to old item and subtract
113      * from new item.
114      CASE UPDATED .AND. item_no <> old_itm
115      new_qty = qty_sold
116      new_itm = item_no
117      SELECT inv
118      SEEK old_itm
119      IF FOUND()
120      | REPLACE onstock WITH onstock + old_qty
121      ENDIF
122      SEEK new_itm
123      IF FOUND()
124      | REPLACE onstock WITH onstock - new_qty
125      ENDIF
126      * If quantity changed, update INV.DBF
127      CASE UPDATED .AND. qty_sold <> old_qty
128      diff = qty_sold - old_qty
129      SELECT inv
130      SEEK old_itm
131      IF FOUND()
132      | REPLACE onstock WITH onstock - diff
133      ENDIF
134      ENDCASE (Updating INV.DBF after editing)
135      mok = " "
136      @ 22,5 SAY "More records to edit? [Y/N] =====>"
137      @ 22,43 GET mok PICTURE "!"
138      READ
139      IF mok = "N"
140      <-----EXIT
141      ENDIF
142      ENDDO (While need more edit)
143      ENDCASE (howmany > 0)
144      ENDIF (minv > 0)
145      ENDDO (While still editing)
146      CLOSE DATABASES
147      RETURN
148
149      *****
150      * End of file SALEDT.PRG *
151      *****

```

يشتمل البرنامج على دوارتين متداخلتين تتكرر الكبرى (الخارجية) حتى يدخل المستفيد رقم صفر ردا على رقم الفاتورة المطلوب معالجة بياناتها. فإذا أدخل رقمًا بخلاف الصفر فإن البرنامج يتحقق من وجود سجلات داخل هذه الفاتورة فإذا لم يجد سجلات لفاتورة بهذا الرقم فمعنى هذا أن رقم هذه الفاتورة خطأ أو غير موجود بالملف. ولذلك تظهر رسالة للمستفيد بهذا المعنى لتصحيح الرقم وإعادة المحاولة أما إذا وجد سجلات لهذه الفاتورة فيتم فتح دارة ثانية (داخلية) وفيها تظهر كل سجلات الفاتورة ليختار المستفيد رقم السجل الذي يرغب في تعديل بياناته وتتكرر الدارة الداخلية طالما أن المستفيد يرغب في تعديل سجل آخر داخل نفس الفاتورة التي اختارها فإذا انتهى من تعديل السجل أو السجلات المطلوبة من الفاتورة التي أدخل رقمها ردا على رسالة الاستمرار بلا (N). فيخرج من الدارة الداخلية ويعود للدورة الخارجية ليطالب بإدخال رقم فاتورة أخرى أو رقم صفر لإنهاء البرنامج.

والآن نعود لمناقشة خطوات البرنامج.

- بعد إنشاء وتخصيص حقول ذاكرة يستدعي البرنامج الإجراء pswrd (سطر ١٥) ومهمته التحقق من صلاحيات الشخص الذي سيستخدم هذا البرنامج (سنناقش هذا الإجراء بعد الانتهاء من مناقشة البرنامج).
- نفتح ملف المبيعات في المنطقة A وملف المخزن مرتبًا تبعًا لأرقام الأصناف في المنطقة B داخل الذاكرة (سطر ١٦ - ١٩).
- يفتح البرنامج دارة تتكرر لإدخال رقم الفاتورة المطلوب تعديل أو حذف بياناتها (سطر ٢٠ - ١٤٤). وفيها يتم الآتي:
- استدعاء إجراء border لرسم الشاشة من جديد وقبول رقم الفاتورة (سطر ٢١ - ٢٤).
- إذا كان رقم الفاتورة يساوي صفر تغلق الدارة وبالتالي البرنامج (سطر ٢٦ - ٢٧).
- إذا كان رقم الفاتورة أكثر من صفر تختار ملف المبيعات (سطر ٣١) ويتم ربط السجلات التي يتطابق رقمها في كلا الملفين (سطر ٣٢) معًا وتلاحظ أننا أشرنا إلى ملف المخازن باستخدام الاسم الرمزي (Aliss) بدلاً من رقم المنطقة في أمر

- SET RELATION بخلاف البرنامج السابق لتوضيح أن كلتا الطريقتين جائزة.
- يبدأ البرنامج في عد السجلات التي تخص رقم الفاتورة ويجب أن تزيد عن صفر وإلا تعتبر الفاتورة غير موجودة وتظهر رسالة للمستخدم (سطر ٣٤ - ٣٨).
 - إذا زاد عدد السجلات في الملف الجديد عن صفر - بعبارة أخرى إذا وجدت سجلات لهذه الفاتورة (سطر ٣٩) تظهر شاشة تحتوي على رقم الفاتورة والسجلات الموجودة بها (سطر ٤٣ - ٥٢).
 - يطلب البرنامج إدخال رقم السجل المطلوب تعديله أو حذفه ولأن المستخدم غير مطالب بحفظ أرقام السجلات داخل الملف فإن رقم السجل وباقي بيانات السجل تظهر في سطر واحد ليعرفها المستخدم ويدخل رقم السجل في حقل ذاكرة اسمه recno (سطر ٥٣).
 - ينتقل المؤشر داخل الملف إلى رقم السجل الذي أدخل (سطر ٥٤)
 - يتم نقل الحقول التي من الممكن تعديلها - إلى حقول بالذاكرة وهي رقم الصنف أو الكمية - لأن باقي حقول السجل مثل اسم البائع أو اسم العميل أو التاريخ لا تمثل أهمية في حالة مردودات المبيعات (سطر ٥٧ - ٥٨).
 - تظهر كل بيانات السجل المطلوب داخل شاشة منظمة ليطلع عليها العميل وتظهر رسالة للمستخدم تسأله عن نوع المعالجة المطلوبة هل هي تعديل (M) أو حذف (D) ويطلب بإدخال أحد حرفين M أو D (سطر ٦١ - ٧٩).
 - إذا أجاب المستخدم في الخطوة السابقة بحرف M أي التعديل تظهر شاشة جديدة بها بيانات حقل رقم الصنف والكمية ويتاح للمستخدم تعديلها. (سطر ٨١ - ٨٨).
 - أما إذا أجاب بحرف D أي الإلغاء فيتم حذف السجل بعد ظهور رسالة للتأكد من رغبة المستخدم في الحذف (سطر ٩٠ - ٩٧). وتلاحظ أيضًا في هذا البرنامج أننا أجّلنا عملية الحذف النهائي لأننا سنخصص أحد برامج صيانة الملفات للقيام بهذه المهمة وسيأتي شرحه في الفصل التاسع إن شاء الله. ولذلك وضعنا علامة *
 - أمام أمر PACK.
 - باتباع الخطوات السابقة تم تعديل أو حذف السجل أو السجلات المطلوبة من

فاتورة معينة. إلا أننا يجب ألا ننسى شيئاً هاماً. وهو أن أرصدة الأصناف بالمخازن تعدلت بطرح كميات المبيعات منها بعد عملية البيع ويجب أن تتعدل الآن لتضاف مردودات المبيعات إلى المخازن لتظل أرصدة حقيقية وتعديل أرصدة المخازن لا يخرج عن واحدة من ثلاث حالات:

.. الحالة الأولى: أن تكون أرصدة المخازن حُدثت بعد عملية البيع - وهذا هو المقصود من شرط UPDATED في الأمر الموجود بسطر ١٠٥ لأن حقل UPDATED يتعدل إلى T. بعد كل عملية تحديث (سنناقش هذا المفهوم في برنامج تحديث المخازن في الفصل التاسع) - أن تكون حدثت وتم إلغاء السجل بالكامل من ملف البيع (سطر ١٠٥). وفي هذه الحالة يتم إضافة الأصناف المباعة إلى رصيد المخازن لأن عملية البيع أصبحت لاغية (سطر ١٠٦ - ١١٠).

- الحالة الثانية: إذا حدث تعديل في رقم الصنف (وأيضاً حُدث ملف المخازن) فمعنى هذا أن صنفاً آخرًا هو الذي بيع ولذلك يجب إضافة الكمية المباعة إلى رصيد الصنف القديم الذي سجل خطأً وطرحها من رصيد الصنف الجديد الذي تعدل لأن الأمر معناه أن البيع تحول من الصنف القديم إلى الصنف الجديد (سطر ١١٣ - ١٢٤).

- الحالة الثالثة: إذا حدث تعديل في كمية المبيعات (وأيضاً حُدث ملف المخازن) يتم حساب الفرق بين الكمية القديمة التي سجلت في فاتورة البيع أول مرة والكمية الجديدة التي أدخلت في حالة التعديل. وطبعاً قد يكون الفرق بين الكميتين سالباً أو موجباً. ويتم طرح الفرق من رصيد الصنف بالمخازن (سطر ١٢٦ - ١٣٢).

التحقق من كلمة السر

يستدعي برنامج SALEDT.PRGI الإجراء الجديد pswid وهو الإجراء الذي يتحقق من كلمة السر قبل إعطاء صلاحيات تعديل ملف البيع وأرصدة المخازن. وقد وضعنا هذا الإجراء ضمن ملف الإجراءات ولم نضعه داخل ملف برنامج مستقل رغم أن هذا الإجراء يستدعيه برنامج SALEDT.PRGI فقط لسبيين. الأول قلة عدد

الأوامر التي يشتمل عليها والثاني أن هذا الإجراء يمكن استدعائه داخل أي برنامج آخر في النظام أو قبل استخدام النظام بأكمله.

ويشتمل شكل ١٣ - ٧ على محتويات الإجراء pswrd ومنه تلاحظ الآتي:

- ينشئ الإجراء عددًا به رقم صفر (سطر ٢٧١) ليعد عدد محاولات إدخال كلمة السر لأن النظام يسمح للمستفيد بخمس محاولات بعدها يغلق البرنامج ويعود إلى البرنامج الرئيسي.

```

270 PROCEDURE pswrd
271 n=0
272 @ 22,05 SAY "Enter password (Up to 8 characters) ==>"
273 DO WHILE .T.
274     SET CONSOLE OFF
275     ACCEPT TO pass
276     pass = UPPER(pass)
277     IF TRIM(pass)="MAGDI"
278         SET CONSOLE ON
279         RETURN
280     ELSE
281         n=n+1
282         IF n=5
283             SET CONSOLE ON
284             CLEAR
285             WAIT " Sorry ... You are not allowed "
286             RETURN TO MASTER
287         ENDIF
288         SET CONSOLE ON
289         @ 22,5 SAY "Password not matched . Try again....."
290         LOOP
291     ENDIF
292 ENDDO
293 RETURN
294

```

شكل ١٣ - ٧ الإجراء pswrd

- تظهر رسالة في سطر ٢٢ تطلب إدخال كلمة السر (سطر ٢٧٢).
- تنشأ دارة لتكرار إدخال كلمة السر ما لم يحصل أحد أمرين، الأول: أن تكون كلمة السر صحيحة وهي هنا كلمة MAGDI أو magdi ، الثاني أن يتجاوز عدد مرات المحاولة ٥ مرات. (سطر ٢٧٣ - ٢٩٢).

- قبل إدخال كلمة السر يتم تعطيل عمل الشاشة (سطر ٢٧٤) والغرض من ذلك ألا يتاح لشخص موجود بجانب المستفيد معرفة كلمة السر ويستخدمها بدون إذن ثم تدخل كلمة السر (سطر ٢٧٥) ويتم تحويلها إلى حروف كبيرة لأن المقارنة ستتم على حروف كبيرة وذلك لإعطاء المستفيد الفرصة لضغط المفاتيح الكبيرة أو الصغيرة لكلمة MAGDI (سطر ٢٧٦).
- إذا كانت كلمة السر صحيحة تعود الشاشة إلى العمل ونخرج من الدوارة ونعود لبرنامج التعديل (سطر ٢٧٧ - ٢٧٩).
- إذا كانت كلمة السر خطأ يزداد العداد بواحد وتعود الشاشة للعمل وتظهر رسالة تفيد أن كلمة السر غير متطابقة وتكرر الدوارة حتى تصل قيمة العداد إلى ٥ . فإذا بلغ عدد المحاولات ٥ مرات تعود الشاشة وتظهر بها رسالة للمستفيد تحبزه بعدم صلاحيته لاستخدام النظام ويتم العودة للبرنامج الرئيسي في النظام وليس لبرنامج التعديل (سطر ٢٨٢ - ٢٨٧).

استخراج تقارير البيع

- الاختيار الثالث في قائمة المبيعات هو المبيعات خلال مدة معينة (راجع شكل ٣ - ٧) والبرنامج الذي يتولى هذه المهمة هو SALPRD.PRG .
- والبرنامج يشبه برنامج PURPRD.PRG الذي شرحناه في الفصل السابق لاستخراج تقارير المشتريات إلا أننا استخدمنا في هذا البرنامج فكرة أخرى غير فكرة إعداد تقرير باستخدام مصمم التقارير واستدعائه للتنفيذ.
- والبرنامج لا يستدعي برامج أو إجراءات أخرى للتنفيذ ويشتمل شكل ١٤ - ٧ على محتويات البرنامج . ونشرها فيما يلي :
- يبدأ البرنامج بإنشاء عداد للصفحات يشتمل على رقم ١ ليكتب رقم صفحة التقرير في بدايتها (سطر ١٠).
 - يطلب البرنامج إدخال تاريخ بداية الفترة وتاريخ نهاية الفترة بطريقة مشابهة لبرنامج استخراج تقارير المشتريات الذي مر بنا في الفصل السابق (سطر ١٥ - ٢٠).

```

1 * -----*
2 * Program.....: salprd.prg*
3 * Called form.: salmenu.prg*
4 * Called form.: dbapp.prg*
5 * Programmer...: Magdi M. Abu Al-Ata*
6 * Date.....: Dec. 25,1990*
7 * Purpose.....: To display sales during aperiod*
8 * -----*
9 STORE SPACE(8) TO sdate,edate
10 mpage = 1
11 SELECT B
12 USE inv INDEX item      && Indexed on item_no
13 SELECT A
14 USE sale
15 @ 24,5 SAY "Enter start date: " get sdate PICTURE "99/99/99"
16 read
17 @ 24,45 SAY "Enter end date: " GET edate PICTURE "99/99/99"
18 READ
19 sdate=ctod(sdate)
20 edate=ctod(edate)
21 mtitle = "Sales during period: " + DTOC(sdate)+" to: " + DTOC(edate)
22 answer = " "
23 @ 24,0
24 @ 24,0 SAY "Repot printing. Please wait"
25 SET HEAD OFF
26 SET DEVICE TO PRINT
27 SET CONSOLE OFF
28 SET RELATION TO item_no INTO B
29 SET FILTER TO date_sold >= sdate .AND. date_sold <= edate
30 GO TOP
31 * Print the invoice
32 DO WHILE .NOT. EOF()
33     @ 0,80/2-(LEN(systitle)/2) SAY systitle
34     @ 1,80/2-(LEN(title2)/2) SAY title2
35     @ 3, 2 SAY "Date: "+ mdate
36     @ 3, 50 SAY "Page No.: "+ LTRIM(STR(mpage,3))
37     @ 5,80/2 - (LEN(mtitle)/2) SAY mtitle
38     @ 6,80/2 - (LEN(mtitle)/2) SAY REPLICATE(CHR(196),LEN(mtitle))
39     @ 8,0 SAY REPLICATE(CHR(196),80)
40     @ 9,1 SAY "Item No.  Description"+SPACE(9)+"Quantity"+ ;
41     SPACE(6)+"Pprice"+SPACE(18)+"Amount"
42     @10,0 SAY REPLICATE(CHR(196),80)
43     DO WHILE .NOT. EOF() .AND. PROW() < 55
44         @ PROW()+1,1 SAY item_no
45         @ PROW(),11 SAY b->desc
46         @ PROW(),30 SAY qty_sold
47         @ PROW(),44 SAY b->price
48         @ PROW(),62 SAY qty_sold*b->price PICTURE "999,999,999.00"
49     SKIP
50     ENDDO
51     mpage = mpage + 1
52     EJECT
53 ENDDO
54 * Say total sales
55 SUM qty_sold*b->price TO bttotal

```

```
55 @ PROW()+2,62 SAY "-----"
56 @ PROW()+1,40 SAY "Total sales"
57 @ PROW(),62 SAY btotat PICTURE "999,999,999.00"
58 @ PROW()+1,62 SAY "=====
59 @ PROW()+1,1 SAY " "      && To clear printer buffer
60 * Reset environment
61 SET HEAD ON
62 SET FILTER TO
63 SET DEVICE TO SCREEN
64 SET CONSOLE ON
65 CLOSE DATABASES
66 RETURN
67 *****
68 * End of file SALPRD.PRG *
69 *****
```

تابع شكل ١٤ - ٧ برنامج SALPRD.PRG

- قبل طباعة التقرير تظهر رسالة للمستفيد تخبره أن التقرير يطبع الآن لأن الشاشة ستعطل أثناء طباعة التقرير (سطر ٢٤).
- قبل طباعة التقرير يتوقف عمل الشاشة وتوجه المخرجات على الطابعة (سطر ٢٦ - ٢٧). ويتم ربط السجلات التي تتشابه أرقام أصنافها في كل من ملف المبيعات وملف المخازن (سطر ٢٨).
- يتم اختيار السجلات التي توافق الفترة المطلوبة فقط (سطر ٢٩).
- تنشأ دوائر لطباعة مقدمة التقرير وتشمل بيانات الشركة ونوع التقرير وأعمدته وتكرر الدوائر حتى تنتهي سجلات الملف ويتم اختيار المواقع المناسبة على الورقة لطباعة هذه البيانات.
- تنشأ دوائر داخلية لتكرار طباعة كل سطر من سطور التقرير وتكرر طالما أن التقرير لم ينته وعدد السطور لم يصل إلى ٥٥ سطرًا (سطر ٤٢ - ٤٩) وفي داخل هذه الدوائر تطبع بيانات السجل الواحد ثم ينتقل المؤشر إلى السجل التالي داخل الملف.
- إذا وصل عدد سطور التقرير ٥٥ سطرًا يزداد عداد الصفحات (سطر ٥٠) ويتم قفز صفحة للأمام (سطر ٥١) وتكرر الدائرة الخارجية حتى تنتهي سجلات الملف.

Arabian Computer Center C.R. 39387				
Date: Tuesday 26,February 1991			Page No.: 1	
Sales during period: 01/02/91 to: 28/02/91				
Item No.	Descreption	Quantity	Pprice	Amount
S-319	Monitor	10	350.00	3,500.00
R-654	Floppy Disk	3	3.00	9.00
S-319	Monitor	4	350.00	1,400.00
R-445	Modem	6	500.00	3,000.00
R-25C	PEGA Card	2	1000.00	2,000.00
R-654	Floppy Disk	50	3.00	150.00
S-319	Monitor	2	350.00	700.00
R-654	Floppy Disk	10	3.00	30.00
R-445	Modem	1	500.00	500.00
R-25C	PEGA Card	3	1000.00	3,000.00
S-319	Monitor	2	350.00	700.00
R-654	Floppy Disk	20	3.00	60.00
R-654	Floppy Disk	2	3.00	6.00
R-25C	PEGA Card	40	1000.00	40,000.00
S-319	Monitor	5	350.00	1,750.00
S-312	Printer	5	2000.00	10,000.00
S-312	Printer	4	2000.00	8,000.00
R-25C	PEGA Card	10	1000.00	10,000.00
S-312	Printer	4	2000.00	8,000.00
S-319	Monitor	20	350.00	7,000.00
S-312	Printer	2	2000.00	4,000.00
R-654	Floppy Disk	1000	3.00	3,000.00
S-319	Monitor	4	350.00	1,400.00
S-319	Monitor	5	350.00	1,750.00
R-654	Floppy Disk	400	3.00	1,200.00
S-312	Printer	3	2000.00	6,000.00
R-25C	PEGA Card	1150	1000.00	1,150,000.00
S-312	Printer	2	2000.00	4,000.00
R-654	Floppy Disk	200	3.00	600.00
S-324	Hard Disk	5	1100.00	5,500.00
R-445	Modem	2	500.00	1,000.00
S-312	Printer	1	2000.00	2,000.00
S-319	Monitor	2	350.00	700.00
S-319	Monitor	3	350.00	1,050.00
R-445	Modem	4	500.00	2,000.00
S-312	Printer	2	2000.00	4,000.00
Total sales				1,288,005.00

شكل ١٥ - ٧ صورة تقرير المبيعات خلال شهر فبراير ١٩٩١ م

- بعد انتهاء سجلات الملف يطبع إجمالي مبيعات الفترة في الموقع المناسب على الورق (سطر ٥٤ - ٥٩) وتعاد القيم التي خصصت بأوامر SET إلى وضعها الطبيعي (سطر ٦١ - ٦٤).

ويشتمل شكل ١٥ - ٧ على شكل صفحة من التقرير الناتج من هذا البرنامج.

الفصل الثامن

نظام حسابات العملاء

Customer Accounts

يشتمل هذا الفصل على نظام حسابات العملاء وتلخص مهمة هذا النظام في مراقبة حسابات العملاء من طريق وظائف إضافة أو تعديل أو حذف أو استرجاع حساب عميل، وتسجيل العمليات المدينة والتسديدات في حسابات العملاء، واستخراج التقارير المطلوبة من العملاء بالإضافة إلى كشف الحساب الشهرية وكشف بالحركة المدينة والدائنة لكل حساب وملخص للحسابات.

وبالرغم من أن هذا النظام يرتبط بالتطبيقات التجارية إلا أنه بإمكانك فصل النظام وتشغيله منفرداً أو بيعه في الأسواق سواء بمصلته أو بعد توفيقه مع رغبات العميل.

ترغب شركة الحاسب العربي في تطوير نظام لمراقبة حسابات العملاء وتحتاج لسجلات وافية عن عملائها تشمل رقم حساب كل عميل واسمه وعنوانه ورقم هاتفه واسم الشركة ورصيد الحساب بالإضافة إلى تقارير وافية عن أي عميل أو عملاء مختارين أو كل عملاء الشركة.

والنظام الذي سنشرحه في هذا الفصل يقوم بالوظائف التالية:

- ١ - إضافة عميل جديد إلى ملف العملاء.
- ٢ - تعديل بيانات عميل إذا دعت الحاجة إلى ذلك.
- ٣ - حذف سجل عميل من الملف لانتهاؤه العمل معه لأي سبب.
- ٤ - استرجاع سجل عميل حذف من الملف.
- ٥ - تسجيل المبالغ التي تستحق على العملاء أو التي يسددونها.
- ٦ - تقارير عن العملاء أو عن عينة مختارة منهم أو حتى عن عميل واحد.
- ٧ - ملصقات للخطابات التي ترسل للعملاء من حين لآخر.
- ٨ - الاستفسار عن حالة أي حساب.
- ٩ - إظهار معاملات العميل المدينة والدائنة على الشاشة أو طباعة تقرير بحركة معاملاته.
- ١٠ - طباعة تقرير مختصر عن أي حساب.
- ١١ - طباعة كشف حساب شهري بمعاملات العميل خلال الشهر ورصيده النهائي.

ويشتمل هذا النظام على برامج أشد تعقيداً وتقارير متباينة وأفكار كثيرة. والهدف من ذلك أن تصبح - بعد الانتهاء من دراسة هذا الفصل - قادراً بإذن الله على تطوير نظم إدارة قواعد البيانات مهما كان تعقيدها. ولذلك ننصح أن تبدأ دراسة هذا النظام بعد الانتهاء من دراسة الأنظمة السابقة.

تصميم ملفات قاعدة البيانات

الخطوة التالية في تطوير النظام هي تصميم ملفات قاعدة البيانات

(Database file) وملفات الفهرسة المتصلة بها. ونظام حسابات العملاء الذي بين أيدينا يستخدم ملفين لقاعدة البيانات الأول هو ملف CUSTMER.DBF ويوضح شكل ١ - ٨ مواصفات ملف العملاء. ولكي تنشئ هذا الملف أدخل الأمر الآتي :

CREATE CUSTMR

ثم أدخل مواصفات الملف الموضحة بشكل ١ - ٨. ويتضح منها أن ملف العملاء يشتمل على البيانات الأساسية للعميل مثل رقم الحساب (CUST_ID) ، واسم

Bytes remaining: 3867

CURSOR <— —> Char: + + Word: Home End Pan: ^_ ^_	INSERT Char: Ins Field: ^N Help: F1	DELETE Char: Del Word: ^Y Field: ^U	Up a field: ↑ Down a field: ↓ Exit/Save: ^End Abort: Esc
--	---	---	---

Field	Field Name	Type	Width	Dec		Field Name	Type	Width	Dec	
1	CUST_ID	Character	4			9	BALANCE	Numeric	7	2
2	COMPANY	Character	28			10	MAXCRDT	Numeric	7	2
3	FNAM	Character	12			11	STMT_DAT	Date	8	
4	MNAM	Character	12			12	STMT_BAL	Numeric	7	2
5	LNAM	Character	12							
6	ADDR	Character	28							
7	CITY	Character	17							
8	PHONE	Character	7							

MODIFY STRUCTURE <<> CUSTMR Field: 1/12

Enter the field name.

Field names begin with a letter and may contain letters, digits and underscores

شكل ١ - ٨ مواصفات ملف العملاء الرئيسي CUSTMR.DBF

الشركة (COMPANY) ، واسم العميل الأول (FNAM) ، واسم الأب (MNAM) ، واسم العائلة (LNAM) ، والعنوان (ADDR) ، والمدينة (CITY) ، ورقم الهاتف . (PHONE) وهي كلها حقول حرفية (وقد وضعنا كل من الاسم الأول واسم الأب واسم العائلة في حقل مستقل لتسهيل ترتيب الملف أو البحث فيه فيما بعد). والرصيد (BALANCE) وأقصى رصيد لحساب العميل (MAXCRDT) وتاريخ آخر كشف حساب أرسل للعميل (STMT_DAT) والرصيد الموجود بآخر كشف حساب

أرسل للعميل (STMT_BAL). ويتم تعديل الحقول الأخيرين عقب استخراج كشف الحساب الشهري شهرياً.

الملف الثاني. ملف تاريخي يشتمل على سجل لكل معاملة مدينة أو دائنة تتم مع العميل. بمعنى أن ملف حسابات العملاء الرئيسي (CUSTMR.DBF) يشتمل على سجل واحد لكل حساب أما هذا الملف فمن الممكن أن يشتمل على أكثر من سجل لنفس الحساب. وفي هذا الملف تسجل المدفوعات والمبالغ التي تستحق على العميل ولأن العميل الواحد يسدد باستمرار فسيشتمل هذا الملف على جميع الحركات المدينة والدائنة الخاصة بالعميل.

ويشتمل شكل ٢ - ٨ على مواصفات ملف العملاء التاريخي CUIHIST.DBF ولكي تنشئ هذا الملف أدخل الأمر الآتي:

. CREATE CUIHIST

Bytes remaining: 3941

CURSOR <-- --> Char: < > Word: Home End Pan: ^+ ^-	INSERT Char: Ins Field: ^N Help: F1	DELETE Char: . Del Word: ^Y Field: ^U	Up a field: ↑ Down a field: ↓ Exit/Save: ^End Abort: Esc
--	---	---	---

Field Name	Type	Width	Dec
1 CUST_ID	Character	4	
2 ID_NO	Character	5	
3 DESC	Character	20	
4 DATE	Date	8	
5 BALANCE	Numeric	7	2
6 DEBIT	Numeric	7	2
7 CREDIT	Numeric	7	2
8 POSTED	Logical	1	

MODIFY STRUCTURE << >> CUIHIST Field: 0/0 Caps

Enter the field name.

Field names begin with a letter and may contain letters, digits and underscores

شكل ٢ - ٨ مواصفات الملف التاريخي CUIHIST.DBF

- ثم أدخل المواصفات الموضحة بشكل ٢ - ٨ . ومنها يتضح أن ملف CUHIST.DBF يشتمل على البيانات التالية :
- رقم الحساب (CUST_ID) ويجب أن يشتمل على نفس المواصفات التي يشتمل عليها رقم الحساب والموجودة بملف العملاء الرئيسي السابق (راجع شكل ١ - ٨) وهي نفس الاسم ونفس النوع ونفس الطول وذلك لأننا سنستخدم هذا الرقم كمفتاح لربط الملفين فيما بعد .
 - رقم المستند (ID_NO) ويقصد به الفاتورة أو الشيك الذي تمت به المعاملة المدينة أو الدائنة .
 - اسم المستند (DESC) .
 - تاريخ المعاملة (DATE) .
 - رصيد الحساب بعد العملية المدينة أو الدائنة التي سجلت (BALANCE) .
 - العملية المدينة (DEBIT) وهي استحقاقات على العميل .
 - العملية الدائنة (CREDIT) وهي تسديدات العميل .
 - حقل منطقي يشتمل على علامة توضح هل اشتمل آخر كشف حساب أرسل للعميل على هذه العملية أم لا (POSTED) .

إنشاء ملفات الفهرسة

لأن البيانات توضع في الملف بترتيب دخولها أول مرة فإذا أردت أن تبحث في الملف أو أن تظهر بيانات العملاء بترتيب مغاير لترتيبها داخل الملف فلا بد من عمل فهرسة (Indexing) لملف قاعدة البيانات . ولأن ملف العملاء قد يشتمل على عدد كبير جدًا من العملاء خصوصًا إذا وضعنا في حسابنا زيادة نشاط الشركة في المستقبل وكثرة عدد العملاء ومناسبة النظام للزيادة المتوقعة فيفضل فهرسة الملف . بعبارة أخرى ترتيب الملف بطريقة تسهل الوصول إلى أي عميل بسهولة وقد يكون هذا الترتيب حسب الاسم أو حسب رقم الحساب . ولأن أسماء العملاء قد تتشابه فلن نستطيع السؤال عن عميل باسمه في حالة طلب بياناته لتعديلها أو الاطلاع عليها . ولذلك سنخصص لكل عميل رقمًا يختلف عن الآخر بحيث لا يتشابه رقمان داخل

الملف. هذا الرقم هو رقم الحساب. وهو الذي سيميز كل عميل عن الآخر داخل الملف. وفي هذا النظام يلزمنا ترتيب ملف العملاء الرئيسي بطريقتين:
أولاً: طبقاً لرقم الحساب ويستخدم هذا الرقم لتمييز السجلات عن بعضها في حالة إدخال بيانات العميل أو الرغبة في تعديلها أو حذفها أو الاستفسار عنها.
الثانية: طبقاً لأبجدية الأسماء ويستخدم الترتيب الأبجدي لإظهار تقارير العملاء فيها بعد.

أما ملف العملاء التاريخي فيلزم ترتيبه طبقاً لرقم الحساب وتاريخ المعاملة لتظهر تواريخ المعاملات مسلسلّة داخل رقم الحساب.

ويوضح شكل ٣ - ٨ أسماء ملفات الفهرسة والحقول التابعة لها والمستخدمّة في النظام.

ملف قاعدة البيانات	حقل الفهرسة	اسم ملف الفهرس	طريقة ترتيب الملف
Database file	Key	Index file	Indexed By
CUSTMR	cust_id	CUSTMR.NDX	طبقاً لرقم الحساب
	fnam + mnam + lnam	NAME.NDX	طبقاً لاسم العميل
CUHIST	cust_id+STR(YEAR(date),4)+ +STR(MONTH(date),2)+ STR(DAY(date),2)	CUHIST.NDX	طبقاً لرقم الحساب والتاريخ

شكل ٣ - ٨ أسماء ملفات الفهرسة والحقول التابعة لها

لإنشاء ملفات الفهرسة المطلوبة أدخل هذه الأوامر بعد الانتهاء من إنشاء
ملفات قاعدة البيانات

- . USE CUSTMR
- . INDEX ON fnam + mnam + lnam To name
- . USE CUHIST

. INDEX ON cust_id + STRL (YEAR (date), 4) + STR (MONTH (date), 2) + STR (DAY (date), 2) TO cuhist

لاحظ أننا استخدمنا في حقل الفهرسة في الأمر الثالث تعبيراً يشتمل على الاسم الأول + اسم الأب + اسم العائلة وذلك لأن الأسماء قد تتشابه . فإذا تكرر اسم محمد مثلاً ٣ مرات فترتب الأسماء الثلاثة تبعاً لأبجدية اسم الأب وبالمثل إذا تشابه اسم الأب ترتب أسماء الآباء طبقاً لأبجديات اسم العائلة والمثال التالي يشتمل على ٣ سجلات مرتبة بهذه الطريقة

MOHAMMAD	GAMAL	ALI
MOHAMMAD	GAMAL	HASAN
MOHAMMAD	NASER	OMAR

ونوضح فيما يلي التعبير المستخدم لفهرسة الملف التاريخي طبقاً لرقم الحساب والتاريخ .

إذا كانت الفهرسة ستم على حقل تاريخي فليست هناك مشكلة ، ولكن المشكلة تظهر إذا كنت سترتب الملف طبقاً لبيانات حقل تاريخي وحقل آخر حرفي . (كما هو الحال في مثالنا هذا) في هذه الحالة لا بد من تحويل الحقل التاريخي إلى حرفي والطريقة التقليدية لذلك هي استخدام الوظيفة DTOC(). ولكن تبقى أيضاً عندنا مشكلة ثانية تنتج هذه المشكلة لأن التاريخ يسجل في ملفات قاعدة البيانات بالشكل الأمريكي هكذا : mm/dd/yy وهذا يعني أن معاملات شهر يناير ستظهر كلها قبل شهر فبراير بصرف النظر عن السنة . ولذلك فإن الترتيب الناتج لن يكون ترتيباً تاريخياً ولحل هذه المشكلة نتبع إحدى طريقتين :

١ - استخراج السنة من التاريخ ثم استخراج الشهر ثم اليوم (كما فعلنا) فمثلاً ١٩٩١/٣/١ سيخصص لها هذا الشكل 19910301 وهذه هي الطريقة التي اخترناها وقد استخدمنا لذلك الوظائف STR() ، YEAR() ، MONTH () ، DAY () .

٢ - استخدام الوظيفة () DTOC لتحويل التاريخ إلى عبارة حرفية . وفي هذه الحالة لابد من استخدام أمر SET DATE ANSI لتحويل التاريخ إلى شكل ANSI وفي هذا الشكل يسجل التاريخ هكذا yy.mm.dd .

ملاحظة : سنشرح طريقة عملية أخرى للفهرسة عند شرح قائمة صيانة الملفات في الفصل التاسع .

تخطيط برامج النظام

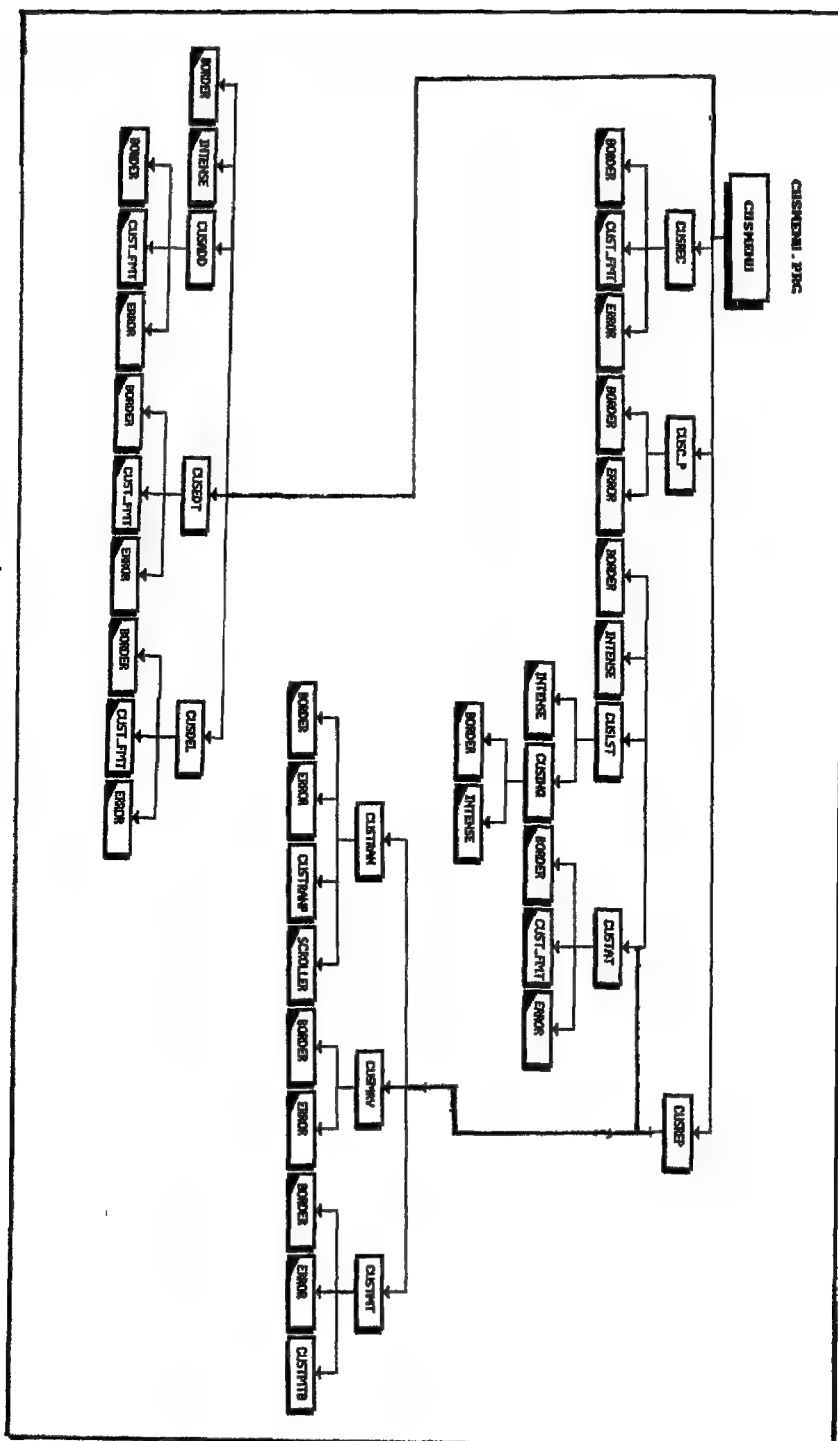
الخطوة التالية في تطوير النظام هي رسم خريطة للنظام توضح البرامج التي يشتمل عليها . ويجب أن تشتمل هذه الخريطة على الوظائف المطلوبة من النظام . وهي التي سنعتمد عليها في تطوير البرامج بعد ذلك .

ويوضح شكل ٤ - ٨ خريطة النظام التي توضح العلاقة بين البرامج التي يشتمل عليها . وفي هذه الخريطة تم تقسيم النظام إلى وظائف أساسية هي : الإضافة (Adding) والتعديل (Modifying) والحذف (Delcting) والاسترجاع (Recalling) وتسجيل المدفوعات والاستحقاقات (Payments & Charges) والتقارير (Reports) ويمثل كل وظيفة من هذه الوظائف مستطيل داخل خريطة النظام وكلها في مستوى واحد .

قائمة نظام حساب العملاء

يجب أن تلبي قائمة نظام حسابات العملاء الوظائف الأساسية الموضحة بخريطة النظام . ويوضح شكل ٥ - ٨ القائمة الرئيسية لنظام حسابات العملاء والاختيارات التي تشتمل عليها . كما يشتمل شكل ٦ - ٨ على برنامج CUSMENU.PRГ اللازم لإظهار هذه القائمة وتنفيذ اختياراتها .

وتشبه فكرة هذا البرنامج والأوامر التي يشتمل عليها برنامج SALMENU اللازم لإظهار القائمة الرئيسية لنظام المبيعات والذي شرحناه في الفصل السابق .



شكل ٨ - ٤ : خروطة نظام حسابات العملاء

Friday 1, March 1991
Time: 12:02:58

Arabian Computer Center
C.R. 39387

Customer menu

- (1) Add new customer (s).....
- (2) Edit existing customer (s)
- (3) Delete customer (s).....
- (4) Undelete customer (s).....
- (5) Charges and Payments.....
- (6) Reports.....
- (G) Go to main menu.....
- (Q) Quit to DOS.....

Select number or colored letter ==>

شكل ٥ - ٨ القائمة الرئيسية لنظام حسابات العملاء

```

1 * ----- *
2 * Program.....: CUSMENU.PRG *
3 * Called from.: DBAPP.PRG *
4 * Programmer..: Magdi M. Abu Al-Ata *
5 * Date.....: Dec. 25,1990 *
6 * Purpose.....: To display customer menu *
7 * ----- *
8 PRIVATE scrttitle,scrfootr,choice,ok
9 scrttitle = "Customer menu"
10 scrfootr = "Select number or colored letter ==>"
11 ok = .T.
12 DO WHILE ok
13     choice = " "
14     DO border WITH scrttitle,scrfootr
15     * Draw menu
16     @10, 5 SAY "(1) Add new customer (s)....."
17     DO intense WITH 10,10,"A"           && Write letter A with
18     && different color
19     @11, 5 SAY "(2) Edit existing customer (s)"
20     DO intense WITH 11,10,"E"
21     @12, 5 SAY "(3) Delete customer (s)....."
22     DO intense WITH 12,10,"D"
23     @13, 5 SAY "(4) Undelete customer (s)....."
24     DO intense WITH 13,10,"U"

```

شكل ٦ - ٨ برنامج CUSMENU.PRG

```

25 @14, 5 SAY "(5) Charges and Payments....."
26 DO intense WITH 14,10,"C"
27 @15, 5 SAY "(6) Reports....."
28 DO intense WITH 15,10,"R"
29 @16, 5 SAY "(G) Go to main menu....."
30 DO intense WITH 16,10,"G"
31 @17, 5 SAY "(Q) Quit to DOS....."
32 DO intense WITH 17,10,"Q"
33 @22,43 GET choice
34 READ
35 * Do selected option
36 DO CASE
37 CASE choice $ "1aA"
38 | DO cusadd && Add new customer
39 CASE choice $ "2Ee"
40 | DO cusedt && Edit an customer
41 CASE choice $ "3dD"
42 | DO cusdel && Delete an customer
43 CASE choice $ "4Uu"
44 | DO cusrec && Recall an customer
45 CASE choice $ "5cC"
46 | DO cusc_p && Recording charges & payments
47 CASE choice $ "6Rr"
48 | DO cusrep && Customer reports
49 CASE UPPER(choice) = "G"
50 | <-----RETURN && Go to Main menu
51 | CASE UPPER(choice) = "Q"
52 | | CLEAR
53 | <=====QUIT && Quit to DOS
54 | OTHERWISE
55 | <-----LOOP
56 | ENDCASE
57 ENDD
58 *****
59 * End of file CUSMENU.PRG *
60 *****

```

تابع شكل ٦ - ٨ برنامج CUSMENU.PRG

ويستخدم نفس الإجراءات التي شرحناها أيضاً (راجع شرح البرنامج والإجراءات في الفصل السابق).

استكمال باقي برامج النظام

بعد تحديد الهدف من النظام وتصميم الملفات ووضع خريطة النظام والبرامج التي تشتمل عليها وبعد كتابة البرنامج الرئيسي للنظام الذي سيظهر القائمة الرئيسية

يجب كتابة البرامج التي وردت بالبرنامج الرئيسي CUSMENU.PRG لتؤدي الوظائف الموجودة بالقائمة الرئيسية.

إضافة حساب عميل جديد لملف العمل.

لورجعنا إلى قائمة النظام في شكل ٥ - ٨ سنجد أن أول اختيار في القائمة هو إضافة حساب عميل جديد. ويشتمل شكل ٧ - ٨ على شاشة إدخال حساب

Friday 1, March 1991
Time: 12:07:04

Arabian Computer Center
C.R. 39387

Customer main data

Customer ID. :			
Customer name :			
Company Name :			
Address :			
City :		Phone :	
Current balance :	0.00	Maximum credit :	0.00
Last stat. date :	/ /	Last stat. balace :	0.00

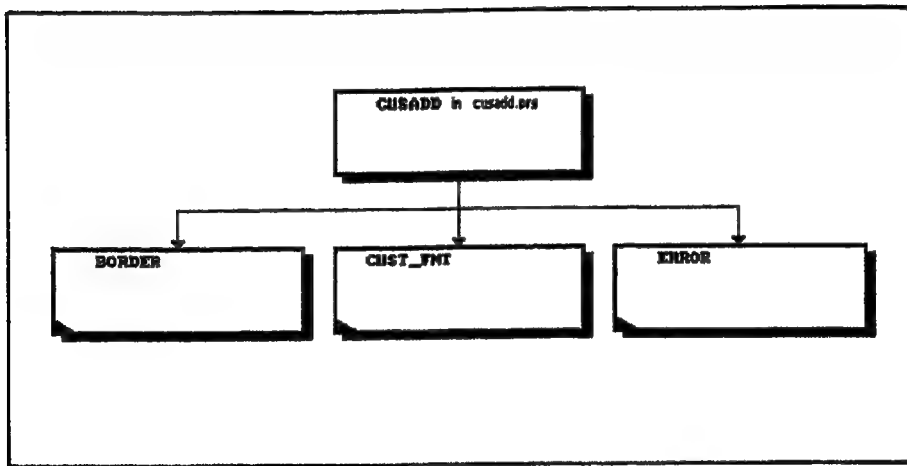
Adding a new customer

Press Enter to return to the menu

شكل ٧ - ٨ شاشة إضافة حساب عميل جديد

عميل جديد والبرنامج اللازم لإضافة عميل جديد إلى الملف هو CUSADD.PRG وتتلخص وظيفته في إدخال بيانات العميل بعد التأكد أن رقم حساب هذا العميل غير موجود من قبل بالملف. ويوضح شكل ٨ - ٨ الخريطة التي توضح علاقة هذا البرنامج بالبرامج الأخرى داخل النظام كما يوضح شكل ٩ - ٨ محتويات هذا البرنامج.

ولا تختلف فكرة هذا البرنامج عن غيره من برامج الإضافة التي شرحناها سابقاً



شكل ٨ - ٨ خريطة برنامج CUSADD.PRG

```

1 * -----*
2 * Program      : CUSADD.PRG                      *
3 * Author       : Magdi M. Abu Al-Ata             *
4 * Date         : October 24, 1990                 *
5 * Purpose      : Adds new customer                *
6 * Called from: CUSMENU.PRG                       *
7 * Called from: DPAPP.PRG                         *
8 * -----*
9 PRIVATE m_custid,m_ok
10 USE custmr INDEX custmr  && Indexed on cust_id
11 DO WHILE .T.          && Loop until exit by entering sapaces in cust_id
12     m_custid = SPACE(4)
13     DO border WITH "Customer main data", "Adding a new customer"
14     DO cust_fmt
15     @24,0
16     @ 24,23 SAY "Press Enter to return to the menu"
17     @ 10,20 GET m_custid PICTURE "@!" && Get customer ID. from the user
18     READ
19     SEEK m_custid          && Lookup account no. in customer file
20     DO CASE
21     CASE m_custid = SPACE(4) && Exit the loop on blank cust_id
22     <-----EXIT
23     CASE FOUND()          && If cust_id already exist
24     | DO error WITH "Customer No. "+TRIM(m_custid)+" already exist."
25     -----LOOP
26     CASE .NOT. FOUND()    && If new cust_id, APPEND record and get data
27     | APPEND BLANK
28     | @ 24,0
29     | REPLACE cust_id with m_custid
    
```

شكل ٨ - ٩ برنامج CUSADD.PRG

```

30      ENDCASE
31      @ 24,0
32      @ 11, 20 GET m_fnam
33      @ 11, 34 GET m_mnam
34      @ 11, 47 GET m_lnam
35      @ 12, 20 GET m_company
36      @ 13, 20 GET m_addr
37      @ 14, 20 GET m_city
38      @ 14, 63 GET m_phone PICTURE "999-9999"
39      @ 17, 24 GET m_bal PICTURE "9,999.99"
40      @ 17, 63 GET m_mxcrdt PICTURE "9,999.99"
41      @ 18, 24 GET m_stdat
42      @ 18, 63 GET m_stbal PICTURE "9,999.99"
43      READ
44      STORE " " TO m_ok
45      * Loop until user enter Y or N in reply message
46      DO WHILE .NOT. m_ok $ "YN"
47      | @ 24,24 SAY "Valid customer data? (Y/N)--->" GET m_ok PICTURE "!"
48      | READ
49      ENDDO
50      IF m_ok = "Y"
51      | @ 24,0
52      | REPLACE company with m_company, fnam with m_fnam, ;
52      | mnam with m_mnam, lnam with m_lnam, addr with m_addr, ;
52      | city with m_city, phone with m_phone
53      | REPLACE balance with m_bal, maxcrdt with m_mxcrdt, ;
53      | stmt_dat with m_stdat, stmt_bal with m_stbal
54      ELSE
55      | DELETE                                && Delete the APPENDED record
56      | ENDIF
57 ENDDO
58 CLOSE DATABASES
59
60 *****
61 * End of file CUSADD.PRg *
62 *****

```

تابع شكل ٩ - ٨ برنامج CUSADD.PRg

ويستدعي البرنامج الإجراء cust_fmt لرسم الشاشة المستخدمة لإدخال وتعديل أو حذف أو استرجاع السجلات. ويشتمل شكل ١٠ - ٨ على محتويات الإجراء cust_fmt وتشبه فكرته فكرة الإجراء ord_fmt الذي شرحناه في الفصل الرابع. وقد قلنا قبل ذلك أنه بإمكانك كتابة البرنامج أمراً أو استخدام أمر:

. CREATE SCREEN cust_fmt

ثم تعديل محتويات ملف cust_fmt.FMT .

```

295 *****
296 * Procedure : cust_fmt
297 * Purpose : Drows customer screen format
298 *****

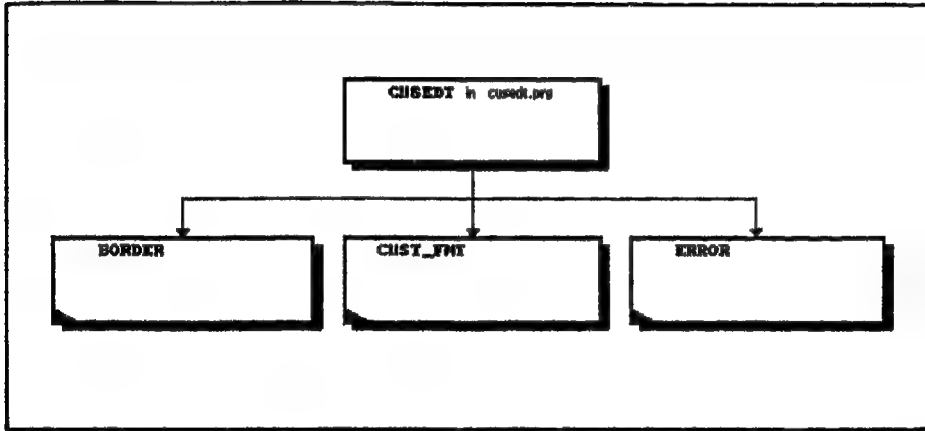
299 PROCEDURE cust_fmt
300 * Make the variables globally available
301 PUBLIC m_cust, m_company, m_addr, m_fnam, m_mnam, m_lnam, m_city, ;
301 m_phone, m_bal, m_mxcrdt, m_stbal, m_stdtd
302 STORE SPACE (04) TO m_cust
303 STORE SPACE (20) TO m_company, m_addr
304 STORE SPACE (12) TO m_fnam, m_mnam, m_lnam
305 STORE SPACE (17) TO m_city
306 STORE SPACE (08) TO m_phone
307 STORE 0 TO m_bal, m_mxcrdt, m_stbal
308 STORE CTOD(" / / ") TO m_stdtd
309 @ 10, 05 SAY "Customer ID. : "
310 @ 10, 20 SAY m_cust
311 @ 11, 05 SAY "Customer name: "
312 @ 11, 20 GET m_fnam
313 @ 11, 34 GET m_mnam
314 @ 11, 47 GET m_lnam
315 @ 12, 05 SAY "Company Name : " GET m_company
316 @ 13, 05 SAY "Address : " GET m_addr
317 @ 14, 05 SAY "City : " GET m_city
318 @ 14, 44 SAY "Phone : " GET m_phone PICTURE "999-9999"
319 @ 17, 05 SAY "Current balance : " GET m_bal PICTURE "9,999.99"
320 @ 17, 44 SAY "Maximum credit : " GET m_mxcrdt PICTURE "9,999.99"
321 @ 18, 05 SAY "Last stmt. date : " GET m_stdtd
322 @ 18, 44 SAY "Last stmt. balace: " GET m_stbal PICTURE "9,999.99"
323 CLEAR GETS
324 RETURN
325

```

شكل ١٠ - ٨ الإجراء *cust_fmt*

تعديل بيانات حساب عميل

نحتاج لتعديل بيانات حساب العميل بعد إدخالها إلى الملف إما لتصحيح خطأ في إدخالها أو لتعديل البيانات بناء على معلومات جديدة عن هذا العميل. والبرنامج الذي يقوم بمهمة تعديل بيانات عميل موجود بالملف هو CUSEDT.PRG. وتتلخص وظيفة هذا البرنامج في تعديل بيانات حساب العميل إذا تأكد أولاً أن رقم حساب العميل موجود بالملف. ويشتمل شكل ١١ - ٨ على الخريطة التي توضح



شكل ١١ - ٨ خريطة برنامج CUSEDT.PRG

علاقة هذا البرنامج بالبرامج الأخرى داخل النظام كما يشتمل شكل ١٢ - ٨ على محتويات هذا البرنامج. وهو يستخدم نفس شاشة إدخال البيانات ولكن بغرض تعديل بيانات العميل بدلاً من إدخال بيانات جديدة (انظر شكل ١٣ - ٨). وتتلخص طريقة عمله في التأكد أولاً أن رقم حساب العميل موجود بالملف ثم إحضار بيانات هذا العميل مع صلاحيات تعديلها. ويستخدم ملف حسابات العملاء والإجراءات الموجودة في برنامج الإضافة السابق CUSADD.PRG ولا تختلف فكرته عن فكرة برامج التعديل السابقة ولذلك لا نرى ضرورة لإعادة شرح محتويات البرنامج هنا.

حذف بيانات عميل من الملف

الاختيار الثالث في قائمة نظام حسابات العملاء هو حذف بيانات عميل موجود بالملف والبرنامج الذي يقوم بعملية الحذف هو CUSDEL.PRG. وهذا البرنامج يتأكد أولاً من وجود رقم حساب العميل في الملف. فإذا وجدته يضع علامة أمام السجل تفيد أن هذا السجل معلم لغرض الحذف. ليتمكن حذفه نهائياً أو استرجاعه حسب الحاجة فيها بعد.

```

1 * -----*
2 * Program      : CUSEDIT.PRG                      *
3 * Author       : Magdi M. Abu Al-Ata              *
4 * Date        : October 24, 1990                  *
5 * Purpose      : To edit an existing customer     *
6 * Called from: CUSMENU.PRG                        *
7 * Called from: DBAPP.PRG                          *
8 * -----*
9 PRIVATE m_custid,m_ok
10 USE custmr INDEX custmr
11 DO WHILE .T.
12     m_custid = SPACE(4)
13     DO border WITH "Customer main data", "Modifying customer data"
14     DO cust_fmt
15     @24,0
16     @ 24,24 SAY "Press Enter to return to the menu"
17     @ 10,20 GET m_custid PICTURE "@!"
18     READ
19     @24,0
20     SEEK m_custid
21     DO CASE
22     CASE m_custid = SPACE(4)
23     <-----EXIT
24     CASE .NOT. FOUND()
25     | DO error WITH "Customer No. " + TRIM(m_custid) + " Not found."
26     <-----LOOP
27     CASE FOUND()
28     | @ 11, 20 GET fnam
29     | @ 11, 34 GET mnam
30     | @ 11, 47 GET lname
31     | @ 12, 20 GET company
32     | @ 13, 20 GET addr
33     | @ 14, 20 GET city
34     | @ 14, 63 GET phone PICTURE "@K 999-9999"
35     | @ 17, 24 GET balance PICTURE "@,999.99"
36     | @ 17, 63 GET maxcrdt PICTURE "@,999.99"
37     | @ 18, 24 GET stmt_dct
38     | @ 19, 63 GET stmt_bal PICTURE "@,999.99"
39     READ
40     ENDCASE
41     * Determine if the operator wants to edit another.
42     STORE " " TO m_ok
43     @ 24, 0
44     DO WHILE .NOT. m_ok $ "YNyn"
45     | @ 24, 20 SAY "Edit another customer record? (Y/N)----> " GET m_ok
46     READ
47     ENDDO
48     IF UPPER (m_ok) = "N"
49     <-----EXIT
50     ENDIF
51     @24,0
52 ENDDO
53 CLOSE DATABASES

```

```

54 RETURN
55
56 *****
57 * End of file CUSEDT.PRG *
58 *****

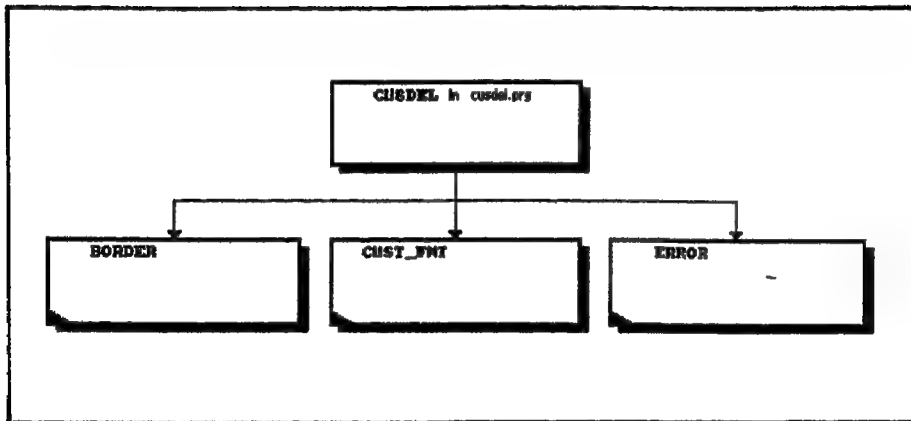
```

تابع شكل ١٢ - ٨ برنامج CUSEDIT.PRG

Friday 1, March 1991		Time: 12:10:00	
Arabian Computer Center C.R. 39387			
Customer main data			
Customer ID. :	0111	Customer name :	Maisara Mahmoud Abu Al-Ata
Company Name :	Abu Al-Ata & Sons	Address :	28 Hegaz Road
City :	Riyadh	Phone :	432-232
Current balance :	600.00	Maximum credit :	3,000.00
Last stat. date :	20/02/91	Last stat. balace :	0.00
			200.00
Modifying customer data			
Edit another customer record? (Y/N)——> <input type="checkbox"/>			

شكل ١٣ - ٨ شاشة تعديل حساب عميل

ويشتمل شكل ١٤ - ٨ على الخريطة التي توضح علاقة هذا البرنامج بالبرامج الأخرى داخل النظام كما يشتمل شكل ١٥ - ٨ على محتويات هذا البرنامج . وهو يستخدم نفس الشاشة المستخدمة لتعديل بيانات عميل مع إلغاء صلاحيات التعديل . وتتلخص طريقة عمله في التأكد أولاً من وجود رقم الحساب بالملف ثم إحضار بيانات العميل على الشاشة إذا وجده ليتأكد المستفيد أن هذه هي البيانات المطلوب حذفها (انظر شكل ١٦ - ٨) . تظهر رسالة للتأكيد تسأل العميل هل هذا هو السجل المطلوب للحذف؟ فإذا أجاب نعم علم السجل لغرض الحذف وإلا بقي



شكل ١٤ - ٨ خريطة برنامج CUSDEL.PRG

```

1 * -----*
2 * Program   : CUSDEL.PRG                      *
3 * Author    : Magdi M. Abu Al-Ata             *
4 * Date      : October 24, 1990                 *
5 * Purpose   : To delete an existing customer  *
6 * Called from: CUSMENU.PRG                    *
7 * Called from: DPAPP.PRG                      *
8 * -----*
9 PRIVATE m_custid,yn,m_ok
10 USE custmr INDEX custmr
11 DO WHILE .T.
12     m_custid = SPACE(4)
13     DO border WITH "Customer main data", "Deleting customer data"
14     DO cust_fmt
15     @24,0
16     @ 24,24 SAY "Press Enter to return to the menu"
17     @ 10,20 GET m_custid PICTURE "@!"
18     READ
19     @24,0
20     SEEK m_custid
21     DO CASE
22     CASE m_custid = space(4)
23     <-----EXIT
24     CASE .NOT. FOUND()
25     | DO error WITH "Customer No. " + TRIM(m_custid) + " Not fond."
26     -----LOOP
27     CASE FOUND()
28     | @ 11, 20 GET fnam
29     | @ 11, 34 GET mnam
30     | @ 11, 47 GET lnam
    
```

شكل ١٥ - ٨ برنامج CUSDEL.PRG

```

31      @ 12, 20 GET company
32      @ 13, 20 GET addr
33      @ 14, 20 GET city
34      @ 14, 63 GET phone PICTURE "AK 999-9999"
35      @ 17, 24 GET balance PICTURE "9,999.99"
36      @ 18, 63 GET maxcrdt PICTURE "9,999.99"
37      @ 18, 24 GET stmt_dat
38      @ 18, 63 GET stmt_bal PICTURE "9,999.99"
39      CLEAR GETS
40      ENDCASE
41      * Confirm deletion
42      STORE " " TO m_ok
43      @ 24,0
44      DO WHILE .NOT. m_ok $ "YNyn"
45      |   @ 24, 21 SAY "Confirm delete this customer? (Y/N)-----> " GET m_ok
46      |   READ
47      ENDDO
48      IF UPPER (m_ok) = "Y"
49      |   DELETE
50      |   ENDIF
51      * Determine if the operator wants to delete another.
52      STORE " " TO yn
53      @ 24, 0
54      DO WHILE .NOT. yn $ "YNyn"
55      |   @ 24, 21 SAY "Delete another customer record? (Y/N)----> " GET yn
56      |   READ
57      ENDDO
58      IF UPPER (yn) = "N"
59      <-----EXIT
60      ENDIF
61      @24,0
62 ENDDO
63 CLOSE DATABASES
64 RETURN
65
66 *****
67 * End of file CUSDEL.PRG *
68 *****

```

تابع شكل ١٥ - ٨ برنامج CUSDEL.PRG

السجل على ما هو عليه . ولا تختلف فكرته عن غيره من برامج الحذف التي شرحناها سابقاً .

استرجاع سجل معلم لفرض الحذف

قد تقرر حذف سجل عميل من الملف ثم تكتشف بعد ذلك حاجتك لبقائه في الملف أو أنك حذفته عملاً غير المقصود ويتيح النظام إمكانية استرجاع سجل

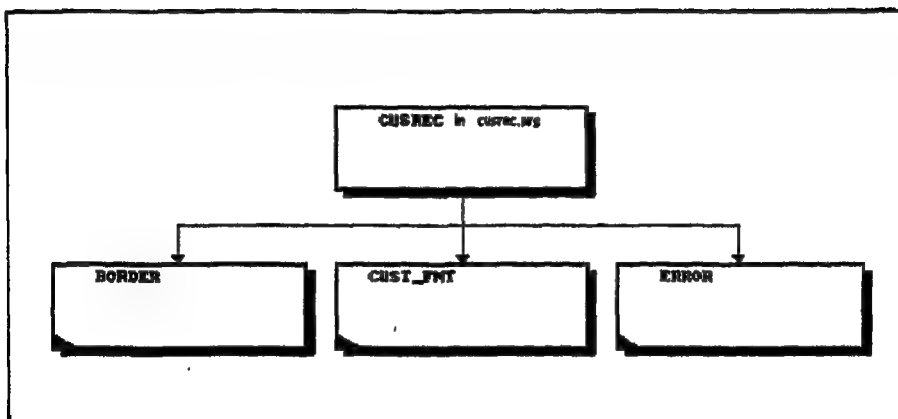
Friday 1, March 1991		Time: 12:14:02	
Arabian Computer Center C.R. 39387			
Customer main data			
Customer ID. :	0128	Customer name :	Gamal Ibrahim Algasim
Company Name :	PC Net. Inc.	Address :	13 King Fahad St.
City :	Dammam	Phone :	542-323
Current balance :	000.00	Maximum credit :	0.00
Last stmt. date :	29/02/91	Last stmt. balance :	1,200.00
Deleting customer data			
Confirm delete this customer? (Y/N)-----> <input type="checkbox"/>			

شكل ١٦ - ٨ شاشة حذف حساب عميل

عُلم بنية حذفه والبرنامج الذي يقوم بهذه العملية هو CUSREC.PRG ويشتمل شكل ١٧ - ٨ على الخريطة التي توضح علاقة هذا البرنامج بالبرامج الأخرى داخل النظام كما يشتمل شكل ١٨ - ٨ على محتويات البرنامج. وهو شبيه ببرنامج الحذف السابق إلا أن الوظيفة الرئيسية للبرنامج تختلف فبينما يستخدم برنامج الحذف أمر DELETE لوضع علامة أمامه السجل توضع أنه مطلوب حذفه يستخدم برنامج الاسترجاع أمر RECALL لاسترجاع المسجل المعلم بنية الحذف ولأن عملية الاسترجاع لا تشكل خطورة مثل عملية الحذف فلا داعي لإظهار رسالة للمستفيد للتأكيد على أن هذا هو السجل المطلوب استرجاعه ما عدا ذلك لا يوجد اختلاف في فكرة هذا البرنامج عن برنامج الحذف ولا في الأوامر التي يشتمل عليها.

تسجيل العمليات المدينة والدائنة

الاختيار الخامس في قائمة حسابات العملاء هو تسجيل العمليات المدينة والدائنة في حسابات العملاء Charges and Payments. والمقصود بالعمليات المدينة



شكل ١٧ - ٨ خريطة برنامج CUSREC.PRG

```

1 * -----*
2 * Program   : CUSREC.PRG                      *
3 * Author    : Magdi M. Abu Al-Ata             *
4 * Date      : October 24, 1990                 *
5 * Purpose   : To delete an existing customer  *
6 * Called from: CUSMENU.PRG                     *
7 * Called from: DPAPP.PRG                       *
8 * -----*
9 PRIVATE m_custid,yn
10 * Because we issue SET DELETED ON in the main procedure, We must
11 * switch it OFF here to appear while SEEKing.
12 SET DELETED OFF
13 USE custmr INDEX custmr
14 DO WHILE .T.
15     m_custid = SPACE(4)
16     DO border WITH "Customer main data", "Undeleting customer data"
17     DO cust_fmt
18     @24,0
19     @ 24,24 SAY "Press Enter to return to the menu"
20     @ 10,20 GET m_custid PICTURE "@1"
21     READ
22     @24,0
23     SEEK m_custid
24     DO CASE
25     CASE m_custid = space(4)
26     <-----EXIT
27     CASE .NOT. FOUND()
28     | DO error WITH "Customer No. " + TRIM(m_custid) + " Not fond."
29     <-----LOOP
30     CASE FOUND()
31     | @ 11, 20 GET fnam
  
```

شكل ١٨ - ٨ برنامج CUSREC.PRG

```

32      @ 11, 34 GET mnam
33      @ 11, 47 GET lnam
34      @ 12, 20 GET company
35      @ 13, 20 GET addr
36      @ 14, 20 GET city
37      @ 14, 63 GET phone PICTURE "999-9999"
38      @ 17, 24 GET balance PICTURE "9,999.99"
39      @ 18, 63 GET maxcrdt PICTURE "9,999.99"
40      @ 18, 24 GET stmt_dat
41      @ 19, 63 GET stmt_bal PICTURE "9,999.99"
42      CLEAR GETS
43      ENDCASE
44      * No need to confirm recalling
45      RECALL
46      * Determine if the operator wants to recall another.
47      STORE " " TO yn
48      @ 24, 0
49      DO WHILE .NOT. yn $ "YNyn"
50      | @ 24, 21 SAY "Recall another customer record? (Y/N)----> " GET yn
51      | READ
52      ENDDO
53      IF UPPER (yn) = "N"
54      <-----EXIT
55      ENDIF
56      @24,0
57 ENDDO
58 CLOSE DATABASES
59 RETURN
60
61 *****
62 * End of file CUSREC.PRG *
63 *****

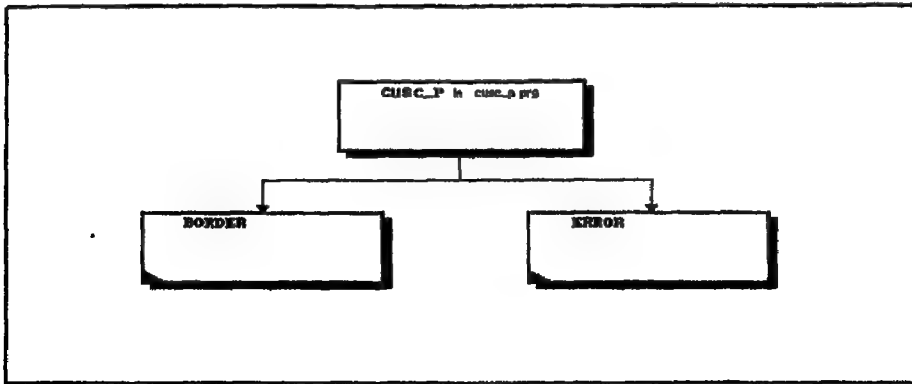
```

تابع شكل ١٨ - ٨

(Charges) العمليات التي تسبب زيادة رصيد العميل المدين مثل البضاعة المباعة له أو الفوائد المدينة. والعمليات الدائنة (Payments) هي العمليات التي تتسبب في نقص رصيد العميل المدين أو زيادة رصيده الدائن ومن أمثلتها التسديدات التي يدفعها.

والبرنامج اللازم لتسجيل العمليات المدينة والدائنة هو CUSC_P.PRG وتتلخص وظيفته في تسجيل الاستحقاقات أو المدفوعات في حساب العميل بعد التأكد من صحة الحساب. وقد استخدمنا في هذا البرنامج طريقة جديدة في السؤال عن السجل. وهي السؤال برقم الحساب أو باسم العميل ويمكن أن تضيف

تسهيلات أخرى للبحث عن السجل بنفس الطريقة مثل اسم الشركة أو اسم العائلة... إلخ. ويشتمل شكل ١٩ - ٨ على الخريطة التي توضح علاقة هذا البرنامج بغيره من البرامج داخل النظام وكما تلاحظ فهو يستدعي إجراءات سبق شرحها كما يشتمل شكل ٢٠ - ٨ على محتويات البرنامج. ولأن البرنامج يشتمل على أفكار جديدة فسنبينها كما يلي:



شكل ١٩ - ٨ خريطة برنامج CUSC_P.PRG

```

1 * -----*
2 * Program      : CUSC_P.PRG                      *
3 * Author       : Magdi M. Abu Al-Ata             *
4 * Date         : October 24, 1990                 *
5 * Purpose      : To enter charges and payments    *
6 * Called from: CUSMENU.PRG                       *
7 * Called from: DBAPP.PRG                         *
8 * -----*
9 PRIVATE scrttitle,scrfootr,m_ok,conf,m_custid,mfirst
10 scrttitle = "Recording charges and payments"
11 scrfootr = "Enter customer ID or customer name, blank to exit"
12 conf = " "
13 DO border with scrttitle,scrfootr
14 * Initialization & open data files
15 SET EXACT OFF      && Because the operator may not knows the exact
16                    && name, we give him the chance to ask by some
17                    && characters and repeat the question in case
18                    && of error.
19 SELECT A
  
```

شكل ٢٠ - ٨ برنامج CUSC_P.PRG

```

20 USE custmr
21 SELECT B
22 USE cuhist
23 * Main loop
24 DO WHILE .T.
25     SELECT custmr
26     @ 10,2 CLEAR TO 20,77
27     m_custid=SPACE(4)
28     mfirst=SPACE(12)
29     * Read customer id or last name
30     @ 12, 06 SAY "Customer ID:" GET m_custid PICTURE "@!"
31     @ 13, 06 SAY "First name: " GET mfirst
32     READ
33     * determine which was entered, and look up in customer file
34     DO CASE
35         * Lookup customer ID
36         CASE m_custid <> SPACE(4)    && Look for non-blank customer ID
37             SET INDEX TO CUSTMR    && If so, index is #1
38             SEEK m_custid
39             * Error if no matches found
40             IF .NOT. FOUND()
41                 DO error WITH "Customer ID "+TRIM(m_custid)+" not found. "
42             --LOOP
43             ENDIF
44         * Lookup customer name
45         CASE mfirst <> SPACE(12)    && Look for non-blank last name
46             SET INDEX TO NAME    && If so, name index is #2
47             * Select matching records
48             SEEK mfirst    && Find last match
49             * Error if no matches found
50             IF .NOT. FOUND()
51                 DO error WITH "Customer name "+TRIM(mfirst)+" not found."
52             --LOOP
53             ENDIF
54         OTHERWISE    && all fields are empty
55     <-----EXIT
56     ENDCASE
57     * Display matching record.
58     @ 17, 04 TO 20,75 DOUBLE
59     @ 18, 06 SAY "Customer ID:" GET cust_id
60     @ 18, 35 SAY "Company : " GET company
61     @ 19, 06 SAY "First name:" GET fnam
62     @ 19, 35 SAY "Last name:" GET lname
63     CLEAR GETS
64     @ 24,0
65     @ 24, 33 SAY "OK (Y or N)?" GET conf PICTURE "!"
66     READ
67     @ 24,0
68     * Check user response
69     IF conf="N"    && If user does not accept
70         @ 17,2 CLEAR TO 20,77
71     --LOOP
72     *
73     ENDIF conf
74     SET INDEX TO CUSTMR

```

```

75 *
76 * Get charges or payments for this account
77 * Append new, blank record, enter today's date
78 SELECT cuhist                                && Use transactions file
79 SET INDEX TO cuhist
80 APPEND BLANK
81 REPLACE cust_id WITH m_custid, date WITH DATE()
81
82 DELETE
83 valid=.F.
84 DO WHILE .not.valid
85     @ 10,2 CLEAR TO 20,77
86     @ 10, 06 SAY "Customer account ID:" GET A->cust_id
87     @ 11, 06 SAY "Last name           :" GET A->lname
88     @ 11, 47 SAY "First name          :" GET A->fname
89     @ 12, 06 SAY "Company              :" GET A->company
90     @ 12, 47 SAY "Transaction date:" GET date
91     @ 14, 2 TO 14,77
92     CLEAR GETS
93     @ 16, 5 SAY "Transaction ID : " GET B->id_no
94     @ 16, 35 SAY "Description      :" GET B->desc
95     @ 18, 5 SAY "Debit (Charges)  :" GET B->debit
96     @ 18, 35 SAY "Credit (Payments):" GET B->credit
97     READ                                && Read Get's
98     * Check data entered. If ok, prompt for acceptance
99     IF debit # 0 .and. credit # 0
100         DO error WITH "Can't accept both debit and a credit."
101     -----LOOP
102     ELSE
103 <-----EXIT
104     ENDIF
105 ENDDO .not.valid
106 * If an amount was entered, update record
107 SELECT custmr
108 REPLACE balance WITH balance + B->debit - B->credit
109 SELECT cuhist
110 REPLACE balance WITH A->balance
111 RECALL                                && Activate finished transaction
112 * Ask about another transaction
113 @ 24, 0
114 m_ok = " "
115 DO WHILE .NOT. m_ok $ "YNyn"
116     @ 24, 20 SAY "Another transaction? (Y/N)-----> " GET m_ok
117     READ
118 ENDDO
119 IF UPPER (m_ok) = "N"
120 <-----EXIT
121     ENDIF
122 @24,0
123 ENDDO .T.                                && End of loop on account
124 SET EXACT ON
125 RELEASE m_custid,mfirst
126 CLOSE DATABASES
127 RETURN
128 *****
129 * End of file CUSC_P.PRG *
130 *****

```

- سطر ١٥ يضع أمر EXACT في حالة OFF وهذه أول مرة منذ بداية الكتاب نستخدم هذا الأمر والهدف من ذلك أن يتاح للمستفيد السؤال بالجزء الذي يعرفه من اسم العميل أو رقمه إذا لم يكن متذكراً باقي الاسم أو رقم الحساب . ولذلك فإن البيانات الأساسية عن السجل تظهر للتأكيد قبل تسجيل المدفوعات والاستحقاقات . وقد أعدنا الأمر إلى حالة ON قبل نهاية البرنامج حتى لا تتأثر به باقي الأنظمة.
- يتم اختيار ملف العملاء الرئيسي وملف المتغيرات التاريخي ووضعهما بالذاكرة (سطر ١٩ - ٢٢).
- تفتح دواة لا نهائية لإدخال حساب العميل يتم الخروج منها إذا أدخل المستفيد فراغات محل رقم الحساب واسم العميل أو إذا أجاب على رسالة الاستمرار في تسجيل معاملة جديدة بلا (سطر ٢٤ - ١٢٣).
- في داخل هذه الدواة يُختار ملف العملاء الرئيسي ليكون هو الحالي وتظهر شاشة صغيرة تستحث العميل لإدخال إما رقم الحساب أو الاسم الأول للعميل المطلوب ويتم قبول أحدهما أو كلاهما (سطر ٢٥ - ٣٢).
- ينفذ البرنامج واحدة من ٣ حالات (سطر ٣٤).
- الأولى: أن تدخل بيانات محل رقم الحساب (سطر ٣٦) وفي هذه الحالة يبحث عن الرقم الذي أدخل في ملف العملاء الرئيسي (سطر ٣٨) فإذا لم يجده (سطر ٤٠) تظهر رسالة تفيد ذلك (سطر ٤١) وينتقل التنفيذ لبداية الدواة ليتمكن المستفيد من إدخال الرقم الصحيح (سطر ٤٢).
- الثانية: أن تدخل بيانات محل الاسم الأول وفي هذه الحالة يبحث عن الاسم الذي أدخل في ملف العملاء الرئيسي فإذا لم يجده تظهر رسالة تفيد ذلك وينتقل التنفيذ لبداية الدواة ليتمكن المستفيد من تكرار المحاولة (سطر ٤٥ - ٥٣).
- الثالثة: أن تدخل فراغات في كلا الحقلين فتغلق الدواة وينتهي البرنامج (سطر ٥٤ - ٥٥).
- تظهر البيانات الأساسية عن الحساب المطلوب مثل رقم الحساب واسم العميل واسم الشركة للتأكيد قبل تسجيل البيانات ومعها رسالة تسأل هل هذه هي بيانات

- السجل المطلوب؟ فإذا كانت الإجابة لا ينتقل التنفيذ لبداية الدوارة لإعطاء المستفيد الفرصة لإدخال بيانات أخرى (سطر ٥٨ - ٧٣).
- إذا رد المستفيد على الرسالة بأن البيانات صحيحة يُختار ملف المتغيرات التاريخي ويضاف سجل خالٍ في نهايته ويتم تعبئة حقول رقم الحساب (cust_id) والتاريخ (date) في هذا السجل (سطر ٧٨ - ٨١). وهذا السجل هو الذي ستسجل فيه العملية المدينة أو الدائنة.
- يدخل البرنامج في دوارة داخلية لإدخال بيانات المعاملة المدينة أو الدائنة (سطر ٨٤ - ١٠٥) ولأن المعاملة الواحدة إما أن تكون مدينة أو دائنة فلا يسمح بتسجيل بيانات في كلا الحقلين لمعاملة واحدة (سطر ٩٩ - ١٠١).
- بعد إدخال المعاملة المدينة أو الدائنة يتعدل رصيد العملاء الموجود في الملف الرئيسي بإضافة مبلغ المعاملة المدينة وخصم مبلغ المعاملة الدائنة (سطر ١٠٧ - ١٠٨).
- ينتقل الرصيد الجديد بعد التعديل إلى ملف المعاملات التاريخي (سطر ١٠٩ - ١١٠).
- تفتح دوارة صغيرة (سطر ١١٥ - ١١٨) لسؤال المستفيد هل يريد تسجيل بيانات معاملة أخرى فإذا أجاب نعم تتكرر الدوارة الكبرى ويستحث لإدخال رقم الحساب أو اسم العميل أما إذا أجاب لا فيتم الخروج من الدوارة وبالتالي البرنامج.

تقارير حسابات العملاء

- الخطوة التالية هي إعداد البرامج التي تقوم بالاستفسارات وطباعة التقارير.
- ويشتمل شكل ٢١ - ٨ على قائمة تقارير حسابات العملاء كما يشتمل شكل ٢٢ - ٨ على البرنامج اللازم لإظهار القائمة وتنفيذ اختياراتها ويوضح شكل ٢٣ - ٨ الخريطة التي توضح علاقة هذا البرنامج بغيره من البرامج داخل النظام.

وبالرجوع إلى قائمة التقارير في شكل ٢١ - ٨ نجد أن التقارير المطلوبة في هذا

النظام هي:

Friday 1, March 1991
Time: 12:18:36

Arabian Computer Center
C.R. 39387

Customer reports menu

- (1) Customer Listing...
- (2) Account(s) Status..
- (3) Account(s) History.
- (4) Account(s) Summary.
- (5) Monthly Statement..
- (G) Go to customer menu
- (Q) Quit to DOS.....

Select number or colored letter ==> █

شكل ٢١ - ٨ قائمة تقارير حسابات العملاء

```

1 * ----- *
2 * Program.....: CUSREP.PRG *
3 * Called form.: CUSMENU.PRG *
4 * Called form.: DBAPP.PRG *
5 * Programmer...: Magdi M. Abu Al-Ata *
6 * Date.....: Dec. 25,1990 *
7 * Purpose.....: To display customer reports menu *
8 * ----- *
9 PRIVATE scrttitle,scrfootr,choice
10 scrttitle = "Customer reports menu"
11 scrfootr = "Select number or colored letter ==>"
12 ok = .T.
13 DO WHILE ok
14     choice = " "
15     DO border WITH scrttitle,scrfootr
16     * Draw menu
17     @10, 5 SAY "(1) Customer Listing..."
18     DO intense WITH 10,10,"C"
19     @11, 5 SAY "(2) Account(s) Status.."
20     DO intense WITH 11,10,"A"
21     @12, 5 SAY "(3) Account(s) History."
22     DO intense WITH 12,21,"H"
23     @13, 5 SAY "(4) Account(s) Summary."
24     DO intense WITH 13,21,"S"

```

تابع شكل ٢٢ - ٨ برنامج CUSREP.PRG

```

25 @14, 5 SAY "(5) Monthly Statement.."
26 DO intense WITH 14,10,"M"
27 @15, 5 SAY "(G) Go to customer menu"
28 DO intense WITH 15,10,"G"
29 @16, 5 SAY "(Q) Quit to DOS....."
30 DO intense WITH 16,10,"Q"
31 @22,43 GET choice
32 READ
33 * Do selected option
34 DO CASE
35 CASE choice $ "1Cc"           && Customer listing
36 | DO custlst
37 CASE choice $ "2Aa"           && Borrowse account staus
38 | DO custat
39 CASE choice $ "3hH"           && View account history
40 | DO custran
41 CASE choice $ "4sS"           && Print account summary
42 | DO cumry
43 CASE choice $ "5Mm"           && Print monthly statemnet
44 | DO custmt
45 CASE UPPER(choice) = "G"       && Go to customer menu
46 <-----RETURN
47 CASE UPPER(choice) = "Q"
48 | CLEAR
49 <=====QUIT                   && Quit the program
50 | OTHERWISE
51 <-----LOOP
52 | ENDCASE
53 ENDD
54 *****
55 * End of file CUSREP.PRG *
56 *****

```

تابع شكل ٢٢ - ٨ برنامج CUSREP.PRG

- ١ - تقرير بيانات العملاء.
 - ٢ - اسفسار عن حالة الحسابات.
 - ٣ - تقرير بحركة حساب معين (المعاملات المدينة والدائنة).
 - ٤ - تقرير بملخص الحسابات.
 - ٥ - طباعة كشوف الحسابات الشهرية للعملاء.
- وسنوضح فيما يلي كيفية استخراج وطباعة هذه التقارير.

تقرير بيانات العملاء

يتيح الاختيار الأول من قائمة تقارير حسابات العملاء طباعة تقارير عن العملاء. ونظراً لكثرة الاختيارات الواردة تحت هذا الاختيار فسنعقد شاشة تابعة تشتمل على أربعة اختيارات للاستفسارات وطباعة تقارير العملاء هي:

- ١ - طباعة تقرير عن كل العملاء مرتب ترتيباً أبجدياً حسب الاسم.
- ٢ - طباعة تقرير عن كل العملاء مرتب حسب أرقام الحسابات.
- ٣ - طباعة تقرير عن عملاء طبقاً لشرط أو شروط معينة.
- ٤ - طباعة ملصقات بريدية تستخدم لكتابة عناوين العملاء عند الحاجة لإرسال كشوف الحسابات الشهرية إليهم.

ويشتمل شكل ٢٤ - ٨ على هذه القائمة الجديدة كما يشتمل شكل ٢٥ - ٨ على برنامج CUSLST.PRG اللازم لإظهار هذه القائمة وتنفيذ اختياراتها وشكل ٢٦ - ٨ على خريطة البرنامج.

Saturday 2, March 1991
Time: 19:41:02

Arabian Computer Center
C.R. 39387

Customer reports menu

(1) Customer Listin
(2) Account(s) Stat
(3) Account(s) Hist
(4) Account(s) Summ
(5) Monthly Stateme
(6) Go to customer
(Q) Quit to DOS....

Reports and labels menu

1. List Alphabetically....
2. List by Account number.
3. List Selected customers
4. Mailing Labels.....
0. Return to customer menu

Be sure your printer ready. Then select option

Select number or col

شكل ٢٤ - ٨ قائمة تقارير العملاء

```

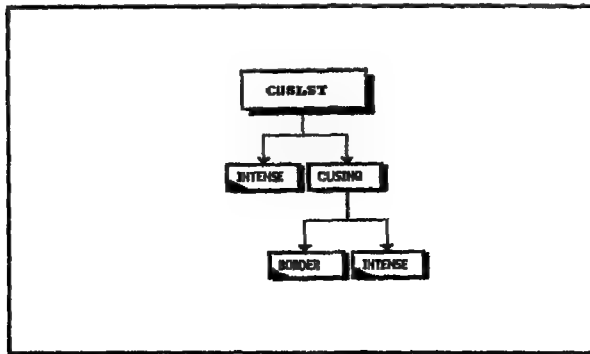
1 * -----*
2 * Program      : CUSLST.PRG                      *
3 * Author       : Magdi M. Abu Al-Ata             *
4 * Date        : October 24, 1990                  *
5 * Purpose     : Print customer accounts listing and labels *
6 * Called from: CUSREP.PRG                         *
7 * Called from: CUSMENU.PRG                       *
8 * Called from: DPAPP.PRG                         *
9 * -----*
10 USE custmr
11 PRIVATE cho
12 DO WHILE .T.
13     cho = " "
14     @ 9,25 CLEAR TO 22,78
15     @ 9,25 TO 22,78 DOUBLE
16     @11,26 TO 11,77
17     @20,26 TO 20,77
18     DO intense WITH 10, 37, "Reports and labels menu"
19     @ 12,30 SAY "1. List Alphabetically...."
20     @ 13,30 SAY "2. List by Account number."
21     @ 14,30 SAY "3. List Selected customers"
22     @ 15,30 SAY "4. Mailing Labels....."
23     @ 16,30 SAY "0. Return to customer menu"
24     DO intense WITH 21,27, "Be sure your printer ready. Then select option"
25     @ 21,74 GET cho
26     READ
27     DO CASE
28     CASE cho = "1"
29         SET INDEX TO name          && Indexed on fname+lname+lname
30         DO intense WITH 21,30, "Alphabetical report printing. Please wait"
31         SET CONSOLE OFF
32         REPORT FORM cusrprt TO PRINT
33         EJECT
34         SET CONSOLE ON
35     CASE cho = "2"
36         SET INDEX TO custmr        && Indexed on acct
37         DO intense WITH 21,27, ;
38         "Serial account number report printing"
39         SET CONSOLE OFF
40         REPORT FORM cusrprt TO PRINT
41         EJECT
42         SET CONSOLE ON
43     CASE cho = "3"
44         DO cusing
45     CASE cho = "4"
46         DO intense WITH 21,27, ;
47         "Labels printing. Please wait"
48         SET CONSOLE OFF
49         LABEL FORM cuslbl TO PRINT
50         EJECT
51         SET CONSOLE ON
52     CASE cho = "0"
53     RETURN
54 
```

```

52 |   ENDCASE
53 ENDDO
54 CLOSE DATABASES
55
56 *****
57 *   End of file CUSLST.PRG   *
58 *****

```

تابع شكل ٢٥ - ٨ برنامج CUSLST.PRG



شكل ٢٦ - ٨ خريطة برنامج CUSLST.PRG

وتتلخص فكرة هذا البرنامج في إظهار قائمة للاستفسارات والتقارير على شكل نافذة داخل قائمة العملاء ولأن «دي بيس ثري بلاس» لا توجد بها إمكانية إظهار نوافذ فقد لجأنا لحذف جزء من الشاشة الذي ستظهر عليه قائمة التقارير قبل إظهارها. وعلى المستخدم أن يختار رقمًا لتنفيذ الاختيار الذي يريده وهو واحد من الاختيارات الأربعة التي ذكرناها قبل قليل أو صفر للخروج من البرنامج والعودة إلى البرنامج السابق. والآن نعود لشرح البرنامج والأفكار التي يشتمل عليها.

- يبدأ البرنامج بفتح ملف العملاء لأنه هو الملف الذي ستستخرج منه التقارير والملصقات. ثم ينشئ دوائر رئيسية مهمتها تكرار تنفيذ اختيارات القائمة التي يظهرها حتى يختار المستخدم الخروج والرجوع إلى القائمة السابقة وفي داخل هذه الدوائر يتم قبول قيمة من المستخدم لتنفيذ الاختيار المطلوب واستدعاء الإجراء

intense لكتابة اسم القائمة ورسالة المحث بلون مخالف لتمييزها عن قائمة العملاء.

- تبدأ الدوارة بإظهار القائمة واختياراتها وقبول قيمة لتحديد الاختيار المطلوب (سطر ١٣ - ٢٥).

- بناء على القيمة التي ستدخل من لوحة المفاتيح يتم تنفيذ أحد الاختيارات وطباعة التقرير المطلوب. ونوضح فيما يلي الاختيارات المتاحة والتقارير التي تنتج عنها.

١ تقرير بيانات العملاء مرتب أبجدياً

ويتم تنفيذه إذا اختار المستفيد الرقم ١ من قائمة التقارير والملصقات (سطر رقم ٢٨) ويتسبب هذا الاختيار في الآتي :

- فتح ملف الفهرس name وهو يظهر البيانات مرتبة ترتيباً أبجدياً حسب الاسم الأول واسم الأب واسم العائلة.

- ظهور رسالة على الشاشة تفيد أن التقرير يطبع الآن أثناء طباعة التقرير على الطابعة حتى لا يشك المستفيد أن البرنامج أو الحاسب تعطل (سطر ٣٠) لأن الشاشة سيتوقف عملها أثناء طباعة التقرير نتيجة لأمر SET CONSOLE OFF لأن التقرير معد أساساً ليظهر على الطابعة.

- استدعاء تقرير معد سلفاً بواسطة مصمم التقارير وطباعته (سطر ٣٢) وبعد الانتهاء من طباعة التقدير نعيد أمر SET CONSOLE إلى وضع ON لتعود الشاشة إلى وضعها الأصلي وهو إظهار المخرجات (سطر ٣٢) ونوضح فيما يلي باختصار كيفية إعداد التقرير بواسطة مصمم التقارير.

ملاحظة: راجع المرجع الأساسي لقاعدة البيانات dBASE III PLUS إذا كنت تريد تفصيلاً أكثر عن استخراج التقارير والملصقات.

لكي تحصل على التقرير المطبوع الموجود في شكل ٢٧ - ٨ اتبع الخطوات

التالية:

Page No.	1	Arabian Computer Center				
01/03/91		C.R. 39387				
		Customer listing				
Cust. ID.	Company Name	Customer Name	Customer Address	Balance	Credit Limit	Last Date
S14	Mahal Computer Co.	Abdullah Ali Abdulaal	Olaia Road	100.00	5000.00	28/02/91
S300	Compag Inc.	Abdullah Megdi Abu Al-Ata	12 Hegaz road	1000.00	9000.00	28/02/91
B150	Al-husaini Inc.	Abdullah Saleh Husaini	12 Teyaran St.	1000.00	7000.00	28/02/91
D500	Computech Inc.	Adel Arafat Bahi Al-Din	Al manar building	1650.00	5000.00	28/02/91
A200	Abu Al-Ata & Sons	Emad Salim Badawi	66 Nahada St.	1450.00	7000.00	28/02/91
B120	PC Net. Inc.	Gamal Ibrahim Algasim	20 King Fahad St.	800.00	8000.00	28/02/91
A111	Abu Al-Ata & Sons	Maisara Mahmoud Abu Al-Ata	20 Hegaz Road	600.00	8000.00	28/02/91
*** Total ***				6600.00		

شكل ٢٧ - تقرير بيانات العملاء مرتب أبجدياً

١ - أدخل الأمر التالي من نقطة المحث بعد فتح ملف CUSTMR.DBF

. CREATE REPORT CUSRPT

ستفتح لك قاعدة البيانات شاشة لتصميم التقرير وبها ٥ اختيارات. اختر option ثم حدد عنوان الصفحة وعرضها وعدد سطورها وحجم الهامش كما هو موضح بشكل ٢٨ - ٨.

٢ - لأننا لا نحتاج لتقسيم التقرير إلى مجموعات انتقل إلى الاختيار Column لاختيار أعمدة التقرير واختر الأعمدة الموضحة بشكل ٢٩ - ٨. اتبع التعليمات التي تظهر تحت سطر الحالة لتوضح لك كيفية اختيار الحقول وكتابة عناوينها. ولاختيار أعمدة التقرير نذكرك بالآتي:

أن اختيار اسم الحقل يتم بضغط مفتاح الإدخال أولاً ثم كتابة اسم الحقل مباشرة أو اضغط مفتاح F10 لاختياره ثم ضغط مفتاح الإدخال مرة ثانية. إن الانتقال إلى الحقل التالي يتم بضغط مفتاح PgDn والحقل السابق بمفتاح .

PgUp

Options	Groups	Columns	Locate	Exit	34:49:13 am
Page title	Arabic				
Page width (positions)	118				
Left margin	2				
Right margin	8				
Lines per page	58				
Double space report	No				
Page eject before printing	No				
Page eject after printing	No				
Plain page	No				

Arabian Computer Center
 C.R. 39387
 Customer listing

CURSOR	← →	Delete char: Del	Insert column: ^N	Insert: Ins
Char: *	→	Delete word: ^T	Report format: F1	Zoom in: ^PgDn
Word: Home End		Delete column: ^U	Abandon: Esc	Zoom out: ^PgUp

MODIFY REPORT <<> CUSRPT.FRM Dpt: 1/9

Enter report title. Exit - Ctrl-End.
 Enter up to four lines of text to be displayed at the top of each report page.

شكل ٢٨ - ٨ الشاشة التي تحدد شكل التقرير

أن أعمدة التقرير يمكن أن تكون محتويات حقل أو نتيجة تعبير مقبول فمثلاً
عامود Customer Name يحتوي على التعبير الآتي ليظهر الاسم الأول واسم
الأب واسم العائلة متجاوزة في عامود واحد.

TRIME (F NAM) + " " + TRIM (MNAME) + " " + LNAME

(انظر شكل ٢٩ - ٨).

Options	Groups	Columns	Locate	Exit	04:54:23 am
			CUST_ID COMPANY trim(CNAME)+ " " + trim(MNAME)+ " " ADDR BALANCE MAXCREDIT START_DATE		
Report Format >>Cust. Company Customer Customer ID. Name Name Address ----- XXXX XXXXXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXXXXX					
ADD NEW REPORT C > REPORT FROM Del: 1/2 Caps					
Position selection bar - ↑↓. Select - ←→. Leave menu - ←→. Step to any column immediately by selecting the column expression.					

شكل ٢٩ - ٨ أعمدة التقرير

٢ تقرير بيانات العملاء مرتب حسب رقم الحساب

يتم تنفيذه إذا اختار المستخدم الرقم ٢ من قائمة التقارير والملصقات. وهو لا
يختلف عن الاختيار الأول إلا في اختيار الفهرس الذي يتحكم في ترتيب الملف
حسب رقم الحساب (سطر ٣٦) ويستخدم نفس التقرير.

٣ تقرير بيانات عملاء مختارين

لأن هذا التقرير يتطلب مجموعة كبيرة من الأوامر فقد فضلنا وضعها في برنامج

مستقل باسم CUSINQ.PRG حتى نحتفظ بالبرنامج الذي بين أيدينا صغيراً. لكن يمكنك إدماج البرنامج الأخير ضمن برنامج التقارير إذا شئت. ويشتمل شكل ٣٠ - ٨ على هذا البرنامج.

```

1 * -----*
2 * Program      : CUSINQ.PRG                      *
3 * Author       : Magdi M. Abu Al-Ata             *
4 * Date         : October 24, 1990                 *
5 * Purpose      : To print customers according some criteria *
6 * Called from: CUSREP.PRG                         *
7 * Called from: CUSMENU.PRG                       *
8 * -----*
9 PRIVATE scrtitle, scrfootr
10 scrtitle = "Listing selected customers"
11 scrfootr = "Select only requested informations. Blanks to exit"
12 DO border WITH scrtitle, scrfootr
13 selected = " "
14 STORE SPACE(04) TO m_acct
15 STORE SPACE(20) TO m_comp, m_addr
16 STORE SPACE(07) TO m_phone
17 STORE SPACE(12) TO m_lnam
18 STORE SPACE(17) TO m_city
19 @ 10, 5 SAY "Account #.." GET m_acct
20 @ 11, 5 SAY "Company...." GET m_comp
21 @ 12, 5 SAY "Phone....." GET m_phone PICTURE "999-9999"
22 @ 13, 5 SAY "Family name" GET m_lnam
23 @ 14, 5 SAY "City....." GET m_city
24 @ 15, 5 SAY "Address...." GET m_addr
25 READ
26 IF m_acct <> SPACE(4)  && If user inquire account no.
27 |   selected = selected + "cust_id = TRIM(m_acct) .AND."
28 ENDIF
29 IF m_comp <> SPACE(20)  && If user inquire company
30 |   selected = selected + "company = TRIM(m_comp) .AND."
31 ENDIF
32 IF m_phone <> SPACE(7)  && If user inquire phone
33 |   selected = selected + "phone = TRIM(m_phone) .AND."
34 ENDIF
35 IF m_lnam <> SPACE(12)  && If user entered last name
36 |   selected = selected + "lnam = TRIM(m_lnam) .AND."
37 ENDIF
38 IF m_city <> SPACE(17)  && If user inquire city
39 |   selected = selected + "city = TRIM(m_city) .AND."
40 ENDIF
41 IF m_addr <> SPACE(20)  && If user inquire address
42 |   selected = selected + "TRIM(m_addr) $ addr .AND."
43 ENDIF
44 * If user didn't enter any criteria
45 IF LEN(TRIM(m_acct+m_comp+m_phone+m_lnam+m_city+m_addr)) = 0

```

```

46 <---RETURN
47 ENDIF
48 selected = SUBSTR(selected,1,LEN(selected) - 5)
49 SET FILTER TO &selected
50 GO TOP
51 IF EOF()          && If new filter has no records
52     DO intense WITH 22, 5, ;
52     "No customers with this criteria. Press any key    "
53     READ
54 ELSE              && If new filter has records
55     DO intense WITH 22, 5, ;
55     "Report printing..... Please wait                "
56     SET CONSOLE OFF
57     REPORT FORM cusrprt TO PRINT
58     EJECT
59     SET CONSOLE ON
60 ENDIF
61 SET FILTER TO
62 RETURN
63
64 *****
65 * End of file CUSINQ.PRG *
66 *****

```

تابع شكل ٣٠ - ٨ برنامج CUSINQ.PRG

ونناقش فيما يلي برنامج CUSINQ الذي يتيح تحديد شرط أو شروط للبيانات التي سيشتمل عليها التقرير.

تحديد شرط / شروط لطباعة بيانات العملاء

يشتمل شكل ٣١ - ٨ على قائمة الشرط التي يمكن تحديدها قبل طباعة التقرير ويمكنك تحديد شرط واحد أو عدة شروط ويتولى البرنامج اختيار العميل أو العملاء الذين ينطبق عليهم الشرط أو الشروط التي حددتها ثم طباعتها. والهدف من قائمة الشروط هو إتاحة الفرصة للمستفيد للاستفسار عن عميل بالمعلومة المتوفرة عنه فقد تكون هذه المعلومة هي رقم الهاتف أو اسم الشركة أو رقم الحساب أو قد تعرف اسم العائلة فقط ولا تعرف باقي الاسم. إلى آخر الشروط التي تظهر بقائمة الشروط. ويمكنك إضافة شروط أخرى أو اختصار هذه الشروط حسب حاجتك. (لاحظ أن بعض هذه الشروط لا يشابه في الملف كله) فمثلاً إذا أردت طباعة كل العملاء في مدينة القاهرة اكتب CAIRO أمام الشرط City أما إذا أردت تحديد أكثر

Tuesday 26, February 1991
Time: 14:52:18

Arabian Computer Center
C.R. 39387

Listing selected customers

Account #..	
Company....	
Phone.....	
Family name	
City.....	
Address....	

Select only requested informations. Blanks to exit

شكل ٣١ - ٨ قائمة شروط السجلات المختارة

من شرط. كما في حالة اختيارك للعملاء الذين يسكنون مدينة القاهرة وفي نفس الوقت اسم عائلاتهم هو أبو العطا. اختر CAIRO أمام الشرط City ثم اختر Abu Al-Ata أمام الشرط Family name . . . وهكذا.

- والآن نعود لمناقشة برنامج CUSINO.PRG الموجود في شكل ٣٠ - ٨
- يبدأ البرنامج برسم إطار الشاشة (سطر ١٢) وتخزين مسافات بحقول في الذاكرة مساوية للبيانات التي ستدخل من شاشة الشروط (سطر ١٤ - ١٨).
- يستخدم أوامر GET...SAY...@ لإظهار شاشة الاستفسارات وقبول الشروط من المستخدم (سطر ١٩ - ٢٥).
- بعد ذلك يبدأ البرنامج في فحص الشروط التي سيحددها المستخدم ويستخدم لذلك أوامر IF وسنلقي الضوء على واحدة منها باعتبار أن الباقي يعمل بنفس الفكرة. أمر (4) SPACE <> IF m_acct (سطر ٢٦) معناه إذا لم يشتمل حقل m_acct على فراغات بعبارة أخرى إذا أدخل المستخدم بيانات في هذا الحقل وهي تعني هنا الشرط الذي سيحدده فإن محتويات حقل Selected وهو حقل ذاكرة ينشأ

بفراغ واحد ستكون كالآتي:

الشرط / الشروط الموجودة من قبل (إذا كان هذا هو أول شرط فإن Selected ستشمل على فراغ) + الشرط الجديد وهو

cust_id = TRIM (m_acct)

ومعناه رقم الحساب المساوي للرقم الذي أدخل كشرط من لوحة المفاتيح + كلمة AND. لتربط هذا الشرط مع الشرط التالي في أمر معقول لدى قاعدة البيانات .
- عند تحديد شرط العنوان (Address) فلا يشترط أن يدخل المستفيد العنوان كاملاً .
فبفرض أنه يتذكر كلمة أو حرف من العنوان فيكفي أن يكتب هذه الكلمة ويبحث البرنامج في كل العناوين التي تشتمل على هذه الكلمة ويظهر عملاءها لذلك استخدمنا علامة \$ بدلا من علامة = (سطر ٤٢).

- إذا لم يدخل المستفيد أي شروط فإنه سيخرج من البرنامج ويرجع إلى القائمة السابقة (سطر ٤٥ - ٤٦).

- لأننا نضيف دائماً كلمة AND. في آخر الأمر فإن قاعدة البيانات ستعترض عليها إذا بقيت بدون تحديد شروط بعدها لذلك يلزم حذف هذه الكلمة دائماً من آخر الأمر وهي تتكون من ٥ حروف (سطر ٤٨).

- العمود الفكري في هذا البرنامج يقع في سطر ٤٩ فالأمر SET FILTER TO يجعل قاعدة البيانات تتعامل مع الملف كما لو كان يشتمل على السجلات المتوافقة مع الشرط الذي يحدده فقط والشرط المحدد هنا هو محتويات حقل Selected وهو يشمل دائماً على الشرط أو الشروط التي حددها المستفيد .

- ننصح دائماً باستخدام أمر GO TOP بعد أمر SET FILTER لنضع المؤشر عند أول سجل في السجلات المختارة (سطر ٥٠).

- بعد أن حددنا الشروط ووضعنا السجلات التي تتطابق مع هذه الشروط في الملف يجب طباعة هذه السجلات ولذلك فإن البرنامج يسأل هل هناك سجلات تتطابق مع الشرط / الشروط المحددة أم لا (سطر ٥١ - ٥٩). فإذا لم توجد سجلات تتطابق مع الشروط فإن المؤشر سيكون في نهاية الملف وبالتالي فإن الوظيفة EOF () ستكون صحيحة (سطر ٥١) وفي هذه الحالة ستظهر رسالة للمستفيد تعلمه

أن الملف لا توجد به سجلات تتطابق مع الشروط التي حددها (سطر ٥٢ - ٥٣).
أما إذا وجدت قاعدة البيانات سجلات تتطابق مع الشروط فإن التقرير سيطلع
بنفس الطريقة التي يطبع بها التقارير في الاختيار الأول والثاني (سطر ٥٤ - ٥٨).
- أمر READ في سطر ٥٤ يسمح بتوقف البرنامج لإعطاء المستفيد الفرصة لقراءة
الرسالة.

٤ طباعة الملصقات

الاختيار الرابع في قائمة التقارير والملصقات هو Mailing Labels ويعني
الملصقات البريدية وهذه الملصقات يستخرجها البرنامج من ملف العملاء وتطبع عادة
على ورق خاص مجهز بالملصقات لتؤخذ بعد طباعتها وتوضع على المظروف دون
حاجة لكتابة العناوين يدوياً أو بالآلة الكاتبة على المظروف كما كان يتم في الماضي.
ويوضح شكل ٣٢ - ٨ شكل بيانات الملصقات الناتجة عن البرنامج وتشتمل السطور
٤٥ - ٤٩ من برنامج CUSLST.PRG (راجع شكل ٢٥ - ٨) على الأوامر التي
تسبب في طباعة هذه الملصقات. وهي شبيهة بالأوامر السابقة في البرنامج ما عدا
أمر LABEL FORM (سطر ٤٧) وهذا الأمر يستدعي ملف ملصقات سبق تصميمه
باسم CUSLBL.LBL.

ونوضح فيما يلي كيفية إنشاء وتصميم ملف الملصقات :

١ - من نقطة المحث أدخل الأمر الآتي بعد فتح ملف العملاء

CREATE LABEL CUSLBL

ستفتح لك قاعدة البيانات شاشة لتصميم شكل الملصقة وبها ٣ اختيارات - اختر
Options ثم اختر ٤٠ لعرض الملصقة (Label width = 40). و ٣٠ لعدد سطورها
(Label hight = 3) وملصقتين بعرض الصفحة (Label across page = 2).

٢ - اختر Contents ثم اختر محتويات الملصقة كما هو موضح بشكل ٣٣ - ٨. وكما
قلنا عن تصميم التقرير أنه يجوز أن تختار حقلاً أو تعبيراً مقبولاً يظهر في التقرير
نقول عن الملصقة. انظر التعبير الآتي الذي سيظهر اسم العميل في أول سطر
من الملصقة

Mr. Abdullah Ali Abdulsal Olaia Road, Riyadh.	Mr. Abdullah Magdi Abu Al-Ata 12 Hegaz road, Riyadh.
Mr. Abdullah Saleh Husaini 12 Tayaran St., Cairo.	Mr. Adel Arafa Bahi Al-Din Al manar building, Cairi.
Mr. Emad Salim Badawi 66 Nahada St., Cairo.	Mr. Gemal Ibrahim Algasim 20 King Fahad St., Dammam.
Mr. Maisara Mahmoud Abu Al-Ata 20 Hegaz Road, Riyadh.	

شكل ٣٢ - ٨ شكل المصققة

Options	Contents	Exit
	Label contents 1: "Mr." + TRIM(FNAM) + " " + TRIM(MNAM) + " " + TRIM(LNAM)	
	2: TRIM(ADDR) + " " + TRIM(CITY) + " " + TRIM(STATE) + " " + TRIM(POSTAL)	
	3: TRIM(CITY) + " " + TRIM(STATE) + " " + TRIM(POSTAL)	

CURSOR: < — >	Delete char: Del	Insert row: ^N	Insert: Ins
Char: + -	Delete word: ^T	Toggle menu: F1	Zoom in: ^PgDn
Word: Home End	Delete row: ^U	Abandon: Esc	Zoom out: ^PgUp

MODIFY LABEL	CUSTBL.LBL	Opt: 1/3
--------------	------------	----------

Position selection bar - f1. Select - < . Leave menu - > .
Enter a field/expression list to be displayed on the indicated label line.

شكل ٣٣ - ٨ شاشة محتويات الملصقة

"Mr." + TRIM (FNAM + " " + TRIM (MNAM) + " " + LNAM

٣ - اختر Exit ثم Save لتحتفظ شكل الملصقة على ملف يمكنك طباعته بأمر .

LABEL FORM

استفسار عن حالة حساب

الاختيار الثاني في قائمة التقارير هو إظهار حالة حساب عميل (Account Status) ويوضح شكل ٣٤ - ٨ شاشة الاستفسار عن حالة الحساب . والبرنامج CUSTAT.PRG ويشتمل شكل ٣٥ - ٨ على خريطة البرنامج وشكل ٣٦ - ٨ على محتويات البرنامج . والبرنامج لايشتمل على أفكار أو أوامر جديدة ويشبه برامج التعديل والحذف التي شرحناها في بداية الفصل .

تقرير بحركة الحساب المدينة والدائنة

الاختيار الثالث في قائمة التقارير لإظهار جميع العمليات المدينة والدائنة التي

Sunday 3, March 1991 Time: 05:19:28

Arabian Computer Center
C.N. 39387

Customer account main data

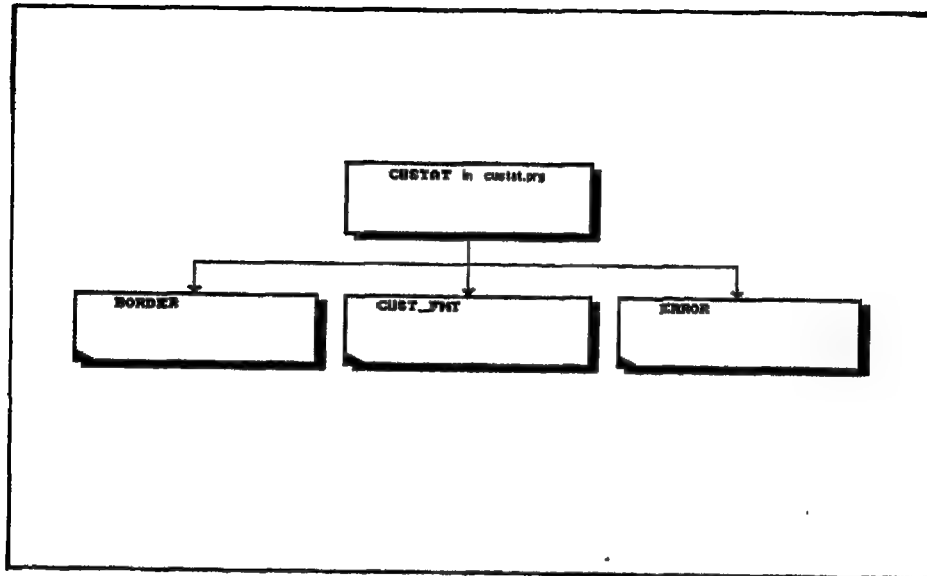
Customer ID. : 0500
Customer name: Abdel [redacted] [redacted] [redacted]
Company Name : Computech Inc.
Address : El mahar building
City : Cairo Phone : 234-223

Current balance : 1,650.00 Maximum credit : 5,000.00
Last stat. date : 01/03/91 Last stat. balace: 1,650.00

Customer account status

Display another Customer account? (Y/N) —> []

شكل ٣٤ - ٨ حالة الحساب رقم D500



شكل ٣٥ - ٨ خريطة برنامج CUSTAT.PRG

```

1 * -----*
2 * Program      : CUSTAT.PRG                      *
3 * Author       : Magdi M. Abu Al-Ata             *
4 * Date        : October 24, 1990                  *
5 * Purpose     : Display customer accounts status  *
6 * Called from: CUSREP.PRG                         *
7 * Called from: CUSMENU.PRG                       *
8 * Called from: DPAPP.PRG                         *
9 * -----*
10 PRIVATE m_custid,m_ok
11 USE custmr INDEX custmr  && Indexed on cust_id
12 DO WHILE .T.
13     m_custid = SPACE(4)
14     DO border WITH ;
14     "Customer account main data", "Customer account status"
15     DO cust_fmt
16     @24,0
17     @ 24,24 SAY "Press Enter to return to the menu"
18     @ 10,20 GET m_custid PICTURE "@1"
19     READ
20     @24,0
21     SEEK m_custid
22     DO CASE
23     CASE m_custid = SPACE(4)
24     <-----EXIT
25     CASE .NOT. FOUND()
26     DO error WITH ;
26     "Customer account No. " + TRIM(m_custid) + " Not found."
27     <-----LOOP
28     CASE FOUND()
29     @ 24,0
30     @ 11, 20 GET fnam
31     @ 11, 34 GET mnam
32     @ 11, 47 GET lnam
33     @ 12, 20 GET company
34     @ 13, 20 GET addr
35     @ 14, 20 GET city
36     @ 14, 63 GET phone PICTURE "##K 999-9999"
37     @ 17, 24 GET balance PICTURE "9,999.99"
38     @ 17, 63 GET maxcrdt PICTURE "9,999.99"
39     @ 18, 24 GET stmt_dat
40     @ 18, 63 GET stmt_bal PICTURE "9,999.99"
41     CLEAR GETS
42     ENDCASE
43     * Determine if the operator wants to display another.
44     STORE " " TO m_ok
45     @ 24, 0
46     DO WHILE .NOT. m_ok $ "YyNn"
47     @ 24, 20 SAY ;
47     "Display another Customer account? (Y/N)-----> " GET m_ok
48     READ
49     ENDDO
50     IF UPPER (m_ok) = "N"
51     <-----EXIT

```

```

52 | ENDIF
53 | @24,0
54 ENDDO
55 CLOSE DATABASES
56 RETURN
57
58 *****
59 * End of file CUSTAT.PRG *
60 *****

```

تابع شكل ٣٦ - ٨ برنامج CUSTAT.PRG

تخص حساب معين (Account Hisotry). والاختيار يظهر حركة الحساب على الشاشة أو يطبع تقرير بحركة الحساب على الطابعة بناء على اختيار المستخدم.

ويشتمل ٣٧ - ٨ على شكل التقرير الذي يظهر على الشاشة بينما يشتمل شكل ٣٨ - ٨ على شكل التقرير المطبوع على الطابعة.

ويشتمل شكل ٣٩ - ٨ على خريطة برنامج CUSTRAN.PRG الذي يتولى إظهار حركة الحساب أو طباعتها. وهو يستدعي برنامج CUSTRANP.PRG إذا طلب

Arabian Computer Center C.R. 39387 Display transactions for an account						
Customer ID. : D500			Company : Computech Inc.			
First name : Adel			Last name: Bahi Al-Din			
Current balance: 1650.00						
Date	ID.	Description	Debit	Credit	Balance	P
01/03/91	12333	test charges	300.00	0.00	300.00	T
01/03/91	54444	Check No. 43242324	0.00	100.00	200.00	T
01/03/91	32222	computer sales	1,000.00	0.00	1,200.00	T
01/03/91	32222	New Sales	450.00	0.00	1,650.00	T

Use F1 for Help, F2 for Print, F3 for End (EOF), or End (EOF), Esc to quit

شكل ٣٧ - ٨ تقرير بحركة الحساب رقم D500 على الشاشة

الفصل الثامن : نظام حسابات العملاء

Friday 1, March 1991

Time: 09:43:00

Arabian Computer Center
C.R. 39387
Customer account transactions

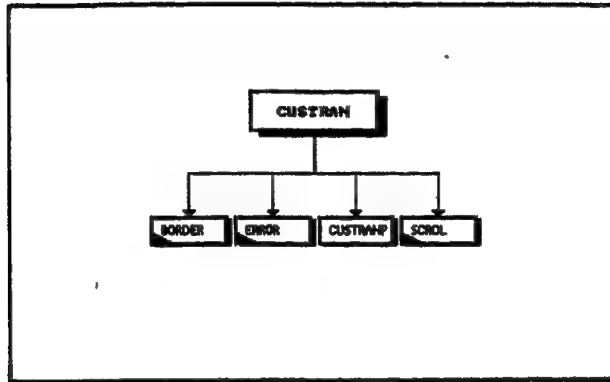
Acct ID: D500
Current balance: 1,650.00
Name: Adel Arafa Bahi Al-Din
Last statement balance: 1,650.00

Company: Computech Inc.
Last statement date: 01/03/91

Date	Id.#	Description	Debit	Credit	Balance	Posted

01/03/91	12333	test charges	300.00		300.00	T
01/03/91	54444	Chech No. 43242324		100.00	200.00	T
01/03/91	32222	computer sales	1,000.00		1,200.00	T
01/03/91	32222	New Sales	450.00		1,650.00	T

شكل ٣٨ - ٨ تقرير مطبوع بحركة الحساب رقم D500



شكل ٣٩ - ٨ خريطة برنامج CUSTRAN.PRG

المستفيد طباعة التقرير على الطابعة. أما برنامج CUTRANP.PRG فلا يستدعي برامج أخرى.

ويشتمل شكل ٤٠ - ٨ على محتويات برنامج CUSTRAN.PRG الذي يظهر التقرير على الشاشة. لأن طباعة التقرير على الطابعة يقوم بها برنامج CUSTRANP.PRG ونناقش فيما يلي كلاً من البرنامجين.

```

1 * -----*
2 * Program   : CUSTRAN.PRG                      *
3 * Author    : Hagdi M. Abu Al-Ata              *
4 * Date      : October 24, 1990                  *
5 * Purpose   : Display transactions for an account *
6 * Called from: CUSREP.PRG                       *
7 * Called from: CUSMENU.PRG                     *
8 * Called from: DPAPP.PRG                       *
9 * -----*
10 PRIVATE scrtitle,scrfootr
11 scrtitle= "View transactions for an account"
12 scrfootr = "Enter customer ID. or press " + CHR(17) + ;
12 CHR(196) + CHR(217) + " to exit."
13 SELECT A
14 USE custmr
15 SELECT B
16 USE cuhist
17 * Set up a loop to enter customer account
18 DO WHILE .T.
    
```

شكل ٤٠ - ٨ برنامج CUSTRAN.PRG

```

19 DO border WITH scrtitle,scrfootr
20 m_custid = SPACE(4)
21 SELECT A
22 SET INDEX TO custmr      && indexed on cust_id
23 @ 14,5 SAY " Customer ID. " GET m_custid PICTURE "@!"
24 READ
25 * Find entered customer id. in main customer file.
26 SEEK m_custid
27 DO CASE
28 CASE m_custid = space(4)
29 <-----EXIT
30 | * if customer id. not found inform the user and try again
31 CASE .NOT. FOUND()
32 | DO error WITH "Customer id. " + rtrim(m_custid) + " not found. "
33 -----LOOP
34 CASE FOUND()
35 @ 24,0
36 scrprt = " "
37 DO WHILE .NOT. scrprt $ "sp"
38 @ 24,20 SAY "Report on Screen or Printer? (S or P) =====> " ;
38 GET scrprt PICTURE "@!"
39 READ
40 ENDDO
41 @ 24,0
42 IF scrprt = "p"
43 DO custranp
44 ELSE      && Need report on screen
45 * Set up screen title
46 CLEAR
47 @ 0,80/2-(LEN(systitle)/2) SAY systitle
48 @ 1,80/2-(LEN(title2)/2) SAY title2
49 @ 2,80/2-(LEN("Display transactions for an account")/2) SAY ;
49 "Display transactions for an account"
50 * Display header
51 @ 4, 2 SAY "Customer ID. : " + cust_id
52 @ 4,45 SAY "Company : " + company
53 @ 5, 2 SAY "First name : " + fnam
54 @ 5,45 SAY "Last name: " + lnam
55 @ 6, 2 SAY "Current balance: " + STR(balance,7,2)
56 @ 7, 0 SAY REPLICATE(CHR(45),79)
57 @ 8, 1 SAY ;
57 "Date ID. Description Debit " ;
57 + "Credit Balance p"
58 @ 09,0 SAY REPLICATE(CHR(45),79)
59 *
60 m_c_line=10      && First line of transaction display
61 m_scrl=11      && Number of lines displayed
62 scnt = 0      && Number of lines to scroll
63 SELECT cuhist
64 SET INDEX TO cuhist .
65 SET FILTER TO cust_id=m_custid
66 GO TOP      && Put pointer at the first record for account
67 * Loop to display transactions for account
68 DO WHILE .T.
69 | c_line = 0      && Counter for lines displayed

```

```

70      @ 10, 1 CLEAR TO 21,78
71      * Loop to display screen lines
72      DO WHILE c_line < m_scl .AND. .NOT. EOF()
73          @ c_line+m_c_line, 1 SAY date
74          @ c_line+m_c_line, 10 SAY id_no
75          @ c_line+m_c_line, 18 SAY desc
76          @ c_line+m_c_line, 39 SAY debit PICTURE "9,999.99"
77          @ c_line+m_c_line, 52 SAY credit PICTURE "9,999.99"
78          @ c_line+m_c_line, 65 SAY balance PICTURE "9,999.99"
79          @ c_line+m_c_line, 78 SAY posted
80          c_line=c_line+1
81          SKIP
82      ENDDO
83      * Display MORE message
84      IF .not. EOF()
85          @ 21, 36 SAY "** MORE **"
86      ENDIF
87      * Wait for one of specified keys
88      DO scrol WITH scnt,c_line,m_scl
89      @ 21, 1 CLEAR TO 22, 78
90      * break out of scroll loop if ESC key pressed
91      IF scnt=999
92          <-----EXIT
93      ENDIF
94      * Position record pointer for next screen
95      SKIP scnt
96      * Handle if at end of file
97      IF EOF()
98          SKIP -1
99      ENDIF
100     ENDDO
101     ENDIF
102     ENDCASE
103 ENDDO
104 SET FILTER TO
105 CLOSE DATABASES
106 RETURN
107
108 *****
109 * End of file CUSTRAN.PRG *
110 *****

```

تابع شكل ٤٠ - ٨ برنامج CUSTRAN.PRG

أولاً: إظهار حركة الحساب على الشاشة

نوضح فيما يلي خطوات البرنامج الموجود في شكل ٤٠ - ٨.

- يبدأ البرنامج بفتح الملفات (سطر ١٣ - ١٦). وإنشاء دوار لا نهائية لإدخال رقم الحساب المطلوب. وتكرر هذه الدوارة ما لم يدخل المستفيد فراغات محل رقم الحساب (سطر ١٨ - ١٠٣).

- في داخل الدوارة الخارجية يرسم البرنامج برواز الشاشة ويطلب إدخال رقم الحساب . فإذا لم يدخل المستفيد فراغات محل رقم الحساب يبحث عنه في ملف العملاء الرئيسي فإذا لم يجده يظهر رسالة للمستفيد تفيد أن رقم الحساب غير موجود ويتيح له الفرصة لإعادة المحاولة أما إذا وجده فتظهر رسالة في سطر ٢٤ من الشاشة تسأله هل يريد التقرير على الشاشة أم على الطابعة؟ (سطر ١٩ - ٤٠).
- الدوارة الصغيرة (سطر ٣٧ - ٤٠) حتى لا يقبل البرنامج من المستفيد سوى حرفين فقط هما S أو P
- إذا اختار المستفيد طباعة التقرير على الطابعة (اختار حرف P) استدعى برنامج الطباعة CUSTRANP.PRГ . أما إذا اختار المستفيد إظهار التقرير على الشاشة (اختار حرف S) فيظهر التقرير على الشاشة طبقاً للخطوات التالية .
- يتم مسح الشاشة لاستقبال التقرير ويكتب عنوان للتقرير يشمل اسم الشركة ونوعية التقرير (سطر ٤٦ - ٤٩).
- تُكتب البيانات الأساسية للحساب المختار (سطر ٥١ - ٥٥).
- يُكتب عنوان أعمدة التقرير (سطر ٥٦ - ٥٨).
- ينشئ حقول الذاكرة التالية (سطر ٦٠ - ٦٢).
- m_c_line : مكان أول سجل سيظهر على الشاشة.
- m_scri : عدد سطور الشاشة.
- sent : يستقبل رقمًا يوضح مقدار طي الشاشة أو الخروج منها.
- يتم اختيار الملف التاريخي واختيار السجلات التي تشتمل على رقم الحساب المطلوب فقط من هذا الملف . ويوضع المؤشر في بداية الملف بعد اختيار السجلات المطلوبة فقط (سطر ٦٣ - ٦٦).
- ينشئ البرنامج دوارة لا نهائية داخل الدوارة الأولى (سطر ٦٨ - ١٠٠) لإظهار سجلات الحساب على الشاشة يتم الخروج منها في حالة واحدة فقط وهي ضغط مفتاح Esc (لاحظ أن التعامل مع مفتاح Esc وغيره من المفاتيح يتم بواسطة الإجراء scroll الذي يستدعيه البرنامج).

- تبدأ هذه الدوارة بتنظيف المساحة التي ستظهر فيها السجلات ثم ينشئ البرنامج دوارة جديدة (ثالثة) يتم الخروج منها إذا انتهت مساحة الشاشة أو انتهت سجلات الملف وتنتهي مساحة الشاشة إذا ظهر عليها ١١ سجلاً لإظهار العمليات المدينة والدائنة التي تخص الحساب على الشاشة مرتبة تاريخياً (سطر ٧٠ - ٨٢). لاحظ أن اختيار مواقع الحقول على الشاشة عملية تنظيمية ترجع إليك.
- لأن الشاشة قد لا تتسع لكل سجلات الحساب المختار فإذا انتهت المساحة المخصصة على الشاشة لإظهار حركة الحساب (١١ سجلاً في الشاشة) ولم تنتهي سجلات الحساب تظهر كلمة *More* في السطر التالي لآخر معاملة (سطر ٨٤ - ٨٦).
- بعد الخروج من الدوارة الثالثة مباشرة (إذا انتهت سجلات الملف أو ظهر ١١ سجلاً على الشاشة) يتم استدعاء الإجراء scrol (سطر ٨٨).
- (سوف نناقش هذا الإجراء بعد الانتهاء من شرح البرنامج).
- جملة IF في سطر ٩١ - ٩٣ تختبر قيمة المتغير scnt الواردة من الإجراء scrol فإذا كانت تساوي ٩٩٩ يتم الخروج من الدوارة الثانية التي تعرض سجلات الحساب والعودة للدوارة الأولى (الخارجية) التي تطلب إدخال رقم حساب جديد. وستعرف من شرح الإجراء scrol أن المتغير scnt يشتمل على الرقم ٩٩٩ في حالة ضغط مفتاح Esc أثناء عرض سجلات الحساب.
- بناء على المفتاح الذي سيضغطه المستفيد (موجود في الإجراء scrol) يتم طوي الشاشة سطر أو صفحة لأعلى أو لأسفل (سطر ٩٥).
- إذا انتهت سجلات الحساب المطلوب ووصل المؤشر لما بعد آخر سجل (نتيجة طي الشاشة) يرجع المؤشر سجلاً واحداً للخلف (سطر ٩٧ - ٩٩) حتى لا تظهر الرسالة التالية:

End of file encountered

ومعناها وصلنا لنهاية الملف.

الإجراء scrol

رغم أننا لم نستدع هذا الإجراء في غير هذا البرنامج إلا أنني فضلت وضعه داخل ملف الإجراءات ليمكنك الاستفادة من فكرته في تطبيقات أخرى. ويشتمل شكل ٤١ - ٨ على محتويات الإجراء scrol ومهمة هذا الإجراء هي قراءة آخر مفتاح يضغطه المستخدم من لوحة المفاتيح بعد عرض أول ١١ سجل من الحساب المختار.

```

326 *****
327 * Procedure : scrol *
328 * Purpose : To accept keystrokes from user to scroll the screen *
329 * Parameters : SCNT - Returns vertical scroll amt to next screen *
330 * C_LINE - Number of lines displayed on last screen *
331 * M_SCRL - Number of lines that fit on a screen *
332 *****

333 PROCEDURE scrol
334 PARAMETERS scnt,c_line,m_scrl
335 PRIVATE i
336 scrlmsg = "Use " + CHR(24)+" "+CHR(25)+" "+ " "+CHR(17)+CHR(196)+ ;
337 CHR(217)+" " Keys, PgUp, PgDn, Home (BOF), " "+or End (EOF); Esc to quit"
338 @ 23, 2 GET scrlmsg
339 i=0
340 SET ESCAPE OFF && Don't interrupt processing on pressing Esc key
341 DO WHILE .not. STR(i,2) $ "13, 5,24,18, 3,27, 1, 6" && Valid keypresses
342 | i=INKEY() && Read keypress
343 ENDDO
344 SET ESCAPE ON && If Esc key pressed, stop processing.
345 * Reset record pointer and cumulative count based on key pressed
346 DO CASE
347 CASE i=27 && ESC key
348 | * flag to Exit the display loop
349 | scnt=999
350 CASE i=5 && UP arrow key
351 | * Move record pointer to one line above start
352 | scnt=-(c_line+1)
353 CASE i=24 && DOWN arrow key
354 | * Move record pointer down one line
355 | scnt=-(c_line-1)
356 CASE i=18 && PgUp key
357 | * Scroll up one screen
358 | scnt=-(c_line+m_scrl)
359 CASE i=3 .OR. i=13 && PgDn or ENTER key
360 | * Scroll down one screen
361 | scnt=0
362 CASE i=1 && Home key

```

شكل ٤١ - ٨ الإجراء scrol

```

363      * Move to beginning of filtered file
364      scnt=0
365      GO TOP
366 CASE i=6          && END key
367      * Move to end of filtered file
368      scnt=-(m_scl-1)
369      GO BOTTOM
370 ENDCASE
371 RETURN
372

```

تابع شكل ٤١ - ٨

وبناء على المفتاح الذي يضغطه المستفيد تطوى الشاشة لأعلى أو لأسفل بمقدار سطر أو شاشة كاملة (١١ سطر) أو يتم تخزين الرقم ٩٩٩ في المتغير scnt مما يسبب الخروج من الدوارة الثانية. والإجراء يقبل ٣ متغيرات من البرنامج (سطر ٣٣٤) هي :
scnt: عبارة عن عداد يحدد مقدار طوي الشاشة.

c_line: رقم يمثل عدد سطور آخر شاشة.

m_scl: رقم يمثل عدد سجلات الشاشة الواحدة.

- يبدأ الإجراء بوضع الرسالة التي ستظهر لتوجه المستفيد لاستخدام مفاتيح الوظائف داخل الذاكرة ثم يظهرها بالألوان العكسية (سطر ٣٣٦ - ٣٣٨) :

- يعطل البرنامج وظيفة مفتاح Esc مؤقتاً أثناء ظهور الرسالة (سطر ٣٤٠) لأنه لا يريد إيقاف البرنامج إذا ضغط المستفيد هذا المفتاح في هذا الموضع بالذات لأنه سيوظف هذا المفتاح للخروج من الدوارة كما سنرى بعد قليل ولكي تستطيع الوظيفة () INKEY قراءة هذا المفتاح.

- ينشئ دوارة صغيرة (سطر ٣٤١ - ٣٤٣) مهمتها ألا يقبل البرنامج من المستفيد إلا المفاتيح الموضحة بالرسالة التي تظهر في آخر الشاشة (راجع شكل ٣٧ - ٨)

وهي ↑, ↓, Enter, PgUp, PgDn, Home, End, Esc

- في داخل هذه الدوارة يستخدم الإجراء الوظيفة () INKEY لقراءة آخر حرف يضغطه المستفيد (سطر ٣٤٢).

- يرجع مفتاح Esc لوظيفته الطبيعية وهي إلغاء البرنامج إذا تم الضغط عليه.

- بعد عرض الأحد عشر سجل الأولى ورسالة التوجيه في أسفل الشاشة (راجع

شكل ٣٧ - ٨) يتم تنفيذ واحدة من الحالات التالية :

❖ أن يضغط المستفيد مفتاح Esc (أي أن $i = 27$) فيتم استبدال محتويات المتغير sent بالرقم ٩٩٩ وعندما يجد البرنامج الرقم ٩٩٩ في المتغير sent سيغلق الدوارة ويرجع إلى الدوارة الأولى (راجع سطر ٩١ - ٩٣ من برنامج CUSTRAN.PRГ).

❖ أن يضغط المستفيد مفتاح السهم ↑ (أي أن $i = 5$) فتتطوي الشاشة لأعلى بمقدار سطر واحد وذلك لأن المؤشر سينتقل داخل الملف بمقدار سجل قبل أول سجل معروض أمامك.

❖ أن يضغط المستفيد مفتاح السهم ↓ (أي أن $i = 24$) فتتطوي الشاشة لأسفل بمقدار سطر واحد وذلك لأن المؤشر سينتقل داخل الملف بمقدار سجل بعد أول سجل معروض أمامك.

❖ أن يضغط المستفيد مفتاح PgUp فتتطوي الشاشة لأعلى بمقدار ١١ سطرًا.

❖ أن يضغط المستفيد مفتاح PgDn أو مفتاح Ender فتتطوي الشاشة بمقدار ١١ سطرًا لأسفل.

❖ أن يضغط المستفيد مفتاح Home فينتقل المؤشر إلى أول سجل بالملف (وهو أول سجل يخص الحساب المختار).

❖ أن يضغط المستفيد مفتاح End فينتقل المؤشر إلى آخر سجل بالملف.

ثانيًا: طباعة حركة الحساب على الطباعة

يشتمل شكل ٤٢ - ٨ على محتويات برنامج CUSTRANP.PRГ اللازم لطباعة التقرير الموجود في شكل ٣٨ - ٨ على الطباعة.

إذا اختار المستفيد في برنامج CUSTRAN.PRГ بعد إدخال رقم الحساب طباعة التقرير على الطباعة يتم استدعاء برنامج CUSTRANP.PRГ وتنفيذه. ونوضح فيما يلي خطوات هذا البرنامج وطريقة عمله:

```

1 * -----*
2 * Program      : CUSTRANP.PRG                      *
3 * Author       : Magdi M. Abu Al-Ata              *
4 * Date        : October 24, 1990                   *
5 * Purpose     : Print transactions for an account  *
6 * Called from: CUSREP.PRG                          *
7 * Called from: CUSMENU.PRG                        *
8 * Called from: DPAPP.PRG                          *
9 * -----*
10 PRIVATE confirm,lines,pageno,ph,mbal
11 * Wait for user to setup printer
12 confirm= ' '
13 @ 24, 23 SAY "Printer ready? (Y/N) =====>" GET confirm PICTURE "I"
14 READ
15 IF .confirm = "N"
16 <---RETURN
17 ENDIF
18 @ 24, 0
19 @ 24, 25 SAY "Report printing.. Please wait"
20 * Select printer for output
21 SET PRINT ON
22 SET CONSOLE OFF
23 SET DEVICE TO PRINT
24 * Print account history
25 mbal=0                                && Initialize running balance
26 pageno=1                             && Initialize page number
27 SELECT cuhist                         && Select transaction file
28 SET INDEX TO cuhist
29 SET FILTER TO cust_id=m_custid        && Select transactions for account
30 GO TOP                                && Find first selected transaction
31 * Loop to display transactions for account
32 DO WHILE .NOT. .EOF()
33
34     * If top of page, print title, account information, headings
35     ph = "Customer account transactions"
36     @ 2, 2 SAY mdate
37     @ 2, 65 SAY "Time: "+TIME()
38     @ 4, 80/2-(LEN(systitle)/2) SAY systitle
39     @ 5, 80/2-(LEN(title2)/2) SAY title2
40     @ 6, 80/2-(LEN(ph)/2) SAY ph
41     @ 7, 80/2-(LEN(ph)/2) SAY REPLICATE(CHR(45),LEN(ph))
42     @ 9, 2 SAY "Acct ID: "
43     @ 9, 11 SAY custmr->cust_id
44     @ 10, 2 SAY "Current balance:"
45     @ 10, 20 SAY custmr->balance.PICTURE '9,999.99'
46     @ 11, 2 SAY "Name: "
47     @ 11, 8 SAY ;
48     TRIM(custmr->fnam)+" "+TRIM(custmr->mnam) +" "+custmr->lnam
49     @ 11, 50 SAY "Company: "
50     @ 11, 59 SAY custmr->company
51     @ 12, 2 SAY "Last statement balance:"
52     @ 12, 25 SAY custmr->stmt_bal PICTURE '9,999.99'
53     @ 12, 50 SAY "Last statement date:"
54     @ 12, 71 SAY custmr->stmt_dat

```

```

54 @ 14, 1 SAY REPLICATE(CHR(45),78)
55 @ 15, 2 SAY "Date      Id.#  Description      " + ;
56 + "Debit    Credit    Balance  Posted"
57 @ 16, 1 SAY REPLICATE(CHR(45),78)
58 lines=17
59
60 * Print next record
61 DO WHILE .NOT. EOF() .and. Lines < 55
62     mbal=mbal+debit-credit      && Calculate running balance
63     @ lines, 2 SAY date
64     @ lines,12 SAY id_no
65     @ lines,19 SAY desc
66     @ lines,41 SAY debit  PICTURE "9,999.99"
67     @ lines,51 SAY credit PICTURE "9,999.99"
68     @ lines,62 SAY mbal   PICTURE "9,999.99"
69     @ lines,73 SAY posted
70     lines=lines+1
71     SKIP
72 ENDDO
73 * If end of page, print page number and eject
74 @ 60, 42 SAY "Page: "+STR(pageno,2)
75 pageno=pageno+1
76 EJECT
77 ENDDO (.NOT. EOF() )
78 * Return control to screen
79 SET DEVICE TO SCREEN
80 SET CONSOLE ON
81 SET PRINT OFF
82 RETURN
83
84 *****
85 * End of file CUSTRANP.PRG *
86 *****

```

تابع شكل ٤٢ - ٨ برنامج CUSTRANP.PRG

- يبدأ البرنامج بإظهار رسالة للمستفيد لتجهيز الطابعة والغرض من هذه الرسالة تجنب رسالة :

Printer not ready

التي تظهر إذا كانت الطابعة مطفأة أو إعطاء المستفيد الفرصة لتجهيز الورق بالطابعة (سطر ١٢ - ١٤).

- إذا رد على الرسالة بما يفيد أن الطابعة غير جاهزة يتم الخروج من البرنامج (سطر ١٥ - ١٧).

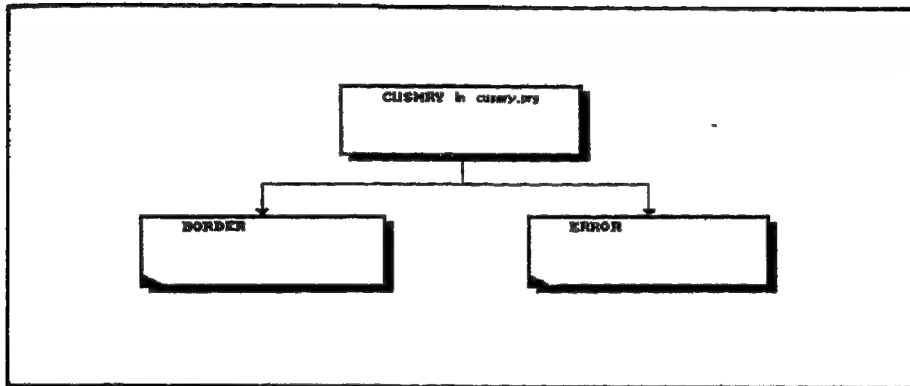
- قبل طباعة التقرير تخرج رسالة تفيد أن التقرير يُطبع لطمأننة المستفيد أن النظام يعمل ثم تعطل الشاشة أثناء طباعة التقرير وتجهز الطابعة لاستقبال التقرير (سطر ١٨ - ٢٣).
- يُختار الملف التاريخي ويُختار فقط السجلات التي تشتمل على رقم الحساب الذي أدخل ويوضع المؤشر في بداية الملف بعد الوضع الجديد (سطر ٢٧ - ٣٠).
- يفتح البرنامج دارة لطباعة التقرير تنتهي بانتهاء سجلات الملف (سطر ٣٢ - ٧٦).
- يبدأ التقرير بطباعة بيانات الشركة واسم التقرير (سطر ٣٥ - ٤١).
- يطبع التقرير البيانات الأساسية عن الحساب المطلوب مثل رقم الحساب واسم العميل ورصيده وتاريخ آخر كشف حساب ورصيد آخر كشف حساب (سطر ٤٢ - ٥٣) ثم يطبع عناوين أعمدة التقرير في الأماكن المناسبة (سطر ٥٤ - ٥٦).
- يفتح البرنامج دارة داخلية لطباعة المعاملات المدينة والدائنة للحساب (سجلات الملف بعد اختيار سجلات هذا الحساب فقط). وهذه الدارة تغلق في حالتين: إذا انتهت سجلات الملف أو إذا وصل عدد سطور الصفحة إلى ٥٥ سطرًا ويطبع التقرير كل سجل في سطر مستقل في المكان المناسب داخل الصفحة (سطر ٦٠ - ٧٠).
- في نهاية الصفحة يكتب رقم الصفحة وتنتقل الطابعة لبداية صفحة جديدة (سطر ٧٣ - ٧٥).
- إذا انتهت طباعة التقرير تعود كل من الشاشة والطابعة إلى وضعها السابق ثم يعود التنفيذ للبرنامج لطلب إدخال رقم حساب آخر.

طباعة ملخص الحسابات

- الاختيار الرابع في قائمة التقارير هو طباعة ملخص الحساب ويوضح شكل ٤٣ - ٨ شكل التقرير الناتج من هذا الاختيار.
- والبرنامج الذي يتولى طباعة التقرير هو CUSMRY.PRГ ويشتمل شكل ٤٤ - ٨ على خريطة البرنامج وشكل ٤٥ - ٨ على محتويات البرنامج.

Friday 1, March 1991	Time: 09:21:48
Arabian Computer Center C.R. 39387 Customer account summary report -----	
Customer ID: D500 -----	
Name : Adel Arafa Bahi Al-Din Company: Computech Inc. Address: Al manar building, Cairi. Phone : 534-223	
Current balance: 1650.00 Credit limit: 5000.00	
Last statement date : 28/02/91 Last statement balance: 0.00	

شكل ٤٣ - ٨ تقرير بملخص الحساب رقم D500



شكل ٤٤ - ٨ خريطة برنامج CUSMRY.PRG

```

1 * -----*
2 * Program   : CUSMRY.PRG                      *
3 * Author    : Magdi M. Abu Al-Ata             *
4 * Date      : October 24, 1990                 *
5 * Purpose   : Print account summary           *
6 * Called from: CUSREP.PRG                     *
7 * Called from: CUSMENU.PRG                   *
8 * Called from: DPAPP.PRG                     *
9 * -----*
10 PRIVATE scrtitle,scrfootr,m_custid
11 scrtitle = "Print account summary"
12 scrfootr = "Enter customer ID. or press " + CHR(17) + ;
12 CHR(196) + CHR(217) + " to exit."
13 SELECT A
14 USE custmr
15 SELECT B
16 USE cuhist
17 * Set up a loop to enter customer account
18 DO WHILE .T.
19     DO border WITH scrtitle,scrfootr
20     m_custid = SPACE(4)
21     SELECT A
22     SET INDEX TO custmr      && indexed on cust_id
23     @ 14,5 SAY " Customer ID. " GET m_custid PICTURE "9!"
24     READ
25     * Find entered customer id. in main customer file.
26     SEEK m_custid
27     DO CASE
28     CASE m_custid = space(4)
29     <-----EXIT
30     | * if customer id. not found inform the user and try again
  
```

شكل ٤٥ - ٨ برنامج CUSMRY.PRG

```

31      CASE .NOT. FOUND()
32      | DO error WITH "Customer id. " + rtrim(m_custid) + " not found. "
33      -----LOOP
34      CASE FOUND()
35      | @ 24,0
36      | p_ok=' '
37      | @ 24, 17 SAY ;
37      | "Setup printer then press any key to continue" GET p_ok
38      | READ
39      | @ 24,0
40      | @ 24, 35 SAY "Printing..."
41      | * Print account summary
42      | hd = "Customer account summary report"
43      | SET DEVICE TO PRINT
44      | SET PRINT ON
45      | SET CONSOLE OFF
46      | @ 1, 2 SAY mdate
47      | @ 1,65 SAY "Time: "+TIME()
48      | @ 3,80/2-(LEN(systitle)/2) SAY systitle
49      | @ 4,80/2-(LEN(title2)/2) SAY title2
50      | @ 5,80/2-(LEN(hd)/2) SAY hd
51      | @ 6,80/2-(LEN(hd)/2) SAY REPLICATE(CHR(45),LEN(hd))
52      | @ 9, 1 SAY "Customer ID: "+custmr->cust_id
53      | @ 10, 1 SAY REPLICATE("-",17)
54      | @ 12, 15 SAY "Name : "+TRIM(custmr->fnam)+" "+ ;
54      | TRIM(custmr->mnam)+" "+custmr->lnam
55      | @ 13, 15 SAY "Company: "+custmr->company
56      | @ 14, 15 SAY "Address: "
57      | @ 14, 27 SAY TRIM(custmr->addr)+" "
58      | @ 15, 27 SAY TRIM(custmr->city)+" "
59      | @ 16, 15 SAY "Phone : "+custmr->phone
60      | @ 18, 1 SAY "Current balance: "+STR(custmr->balance,7,2)
61      | @ 19, 1 SAY "Credit limit : "+STR(custmr->maxcrdt,7,2)
62      | @ 21, 1 SAY "Last statement date : "+ DTOC(custmr->stmt_dat)
63      | @ 22, 1 SAY ;
63      | "Last statement balance: "+ STR(custmr->stmt_bal,7,2)
64      | EJECT
65      | SET DEVICE TO SCREEN
66      | SET PRINT OFF
67      | SET CONSOLE ON
68      | ENDCASE
69      | ENDDO
70      | CLOSE DATABASES
71      | RETURN
72
73      *****
74      * End of file CUSMRY.PRG *
75      *****

```

تابع شكل ٤٥ - ٨ برنامج CUSMRY.PRG

والبرنامج مباشر ولا يشتمل على أفكار أو أوامر تزيد عن الواردة ببرامج طباعة التقارير التي تقدمت والفرق الوحيد بين هذا التقرير والتقرير السابق هو في مواقع البيانات التي يشتمل عليها. فيتم اختيار الأماكن المناسبة لطباعة اسم الشركة وعنوانها والبيانات الأساسية عن الحساب المطلوب. ويكتفي هذا التقرير بطباعة رصيد الحساب فقط ولا يشتمل على تفاصيل معاملات الحساب.

طباعة كشف الحساب الشهري

التقرير الأخير في قائمة التقارير هو طباعة كشف الحساب الشهري (Monthly Statement) ويشتمل شكل ٤٦ - ٨ على تقرير كشف الحساب الشهري للحساب رقم D500 والبرنامج اللازم لاستخراج هذا التقرير وطابعته هو SUSTMT.PRG .

ويشتمل شكل ٤٧ - ٨ على خريطة البرنامج كما يشتمل ٤٨ - ٨ على محتويات البرنامج والبرنامج شبيه ببرنامج طباعة حركة الحساب ويزيد عليه بعض البيانات التي تناسب التقرير الشهري ويتطلب بعض التعديلات في الملفات. وفيما يلي تلقي الضوء على طريقة عمل البرنامج والأفكار التي يشتمل عليها.

- يفتح الملفات المطلوبة ويفتح دوائر لتكرار قبول رقم الحساب من المستفيد وطباعة كشف الحساب المطلوب وتغلق هذه الدوائر إذا أدخلت فراغات محل رقم الحساب وفي هذه الدوائر يتم البحث عن رقم الحساب كما يحدث في باقي البرامج. (سطر ١٣ - ٣٣).

- إذا وجد البرنامج رقم الحساب بملف العملاء الرئيسي يفتح دائرة داخلية لقبول تاريخ كشف الحساب. وهذه الدائرة يتم الخروج منها إذا أجاب المستفيد أن التاريخ الذي أدخله مضبوط أما إذا أجاب أن التاريخ غير مضبوط فتتكرر الدائرة لإدخال التاريخ السليم وقبوله (سطر ٣٧ - ٥٠).

- ويظهر البرنامج للمستفيد إما التاريخ المسجل بالحاسب في بداية اليوم أو تاريخ آخر كشف حساب موجود بملف العملاء الرئيسي أيهما أكبر (سطر ٣٨) ويشترط لقبول التاريخ ألا يقل عن تاريخ آخر كشف حساب أرسل للعميل (سطر ٣٨ - ٤١).

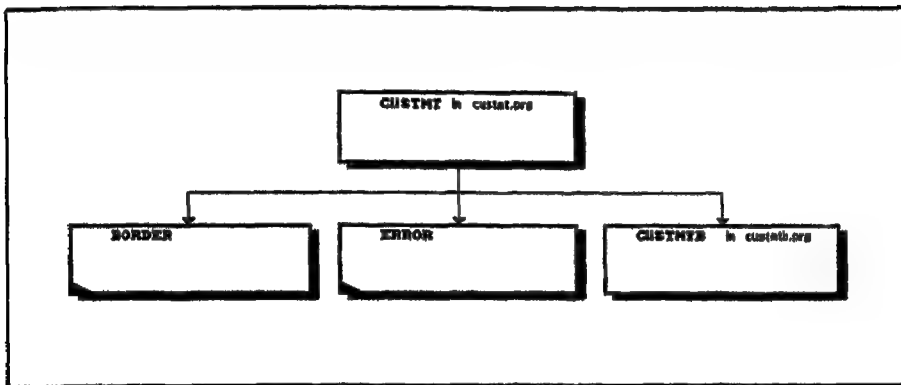
Friday 1, March 1991		Time: 09:35:20	
Arabian Computer Center C.R. 39387 Statement of account -----			
Account ID: D500			
Adel Arafa Bahi Al-Din Computech Inc. Company, Al manar building, Cairi.			
Date	Description	Debit	Credit
			Balance
28/02/91	**Previous balance**		0.00
01/03/91	test charges	300.00	
01/03/91	Chech No. 43242324		-100.00
01/03/91	computer sales	1,000.00	
01/03/91	New Sales	450.00	
			1,650.00
ACCOUNT SUMMARY			

Previous balance			0.00
Payments received			-100.00
Current month charges			1,750.00

CURRENT BALANCE			1,650.00

Please pay your balance before 31/03/91 *****			

شكل ٤٦ - ٨ كشف الحساب الشهري



شكل ٤٧ - ٨ خريطة برنامج CUSTMT.PRG

```

1 * -----*
2 * Program   : CUSTMT.PRG                      *
3 * Author    : Magdi M. Abu Al-Ata             *
4 * Date      : October 24, 1990                 *
5 * Purpose   : Print current month statement for an account *
6 * Called from: CUSREP.PRG                      *
7 * Called from: CUSMENU.PRG                    *
8 * Called from: DPAPP.PRG                      *
9 * -----*
10 PRIVATE scrtitle,scrfootr,m_custid,m_date,m_ok
11 scrtitle = "Print current account statemnet"
12 scrfootr = "Enter customer ID. or press " + ;
13   CHR(17) + CHR(196) + CHR(217) + " to exit."
14 SELECT A
15 USE custmr
16 SELECT B
17 USE cuhist
18 * Set up a loop to enter customer account
19 DO WHILE .T.
20   DO border WITH scrtitle,scrfootr
21   m_custid = SPACE(4)
22   SELECT A
23   SET INDEX TO custmr      && indexed on cust_id
24   @ 14,5 SAY " Customer ID. " GET m_custid PICTURE "@!"
25   READ
26   * Find entered customer id. in main customer file.
27   SEEK m_custid
28   DO CASE
29     CASE m_custid = space(4)
30     <-----EXIT
31     * if customer id. not found inform the user and try again
32     CASE .NOT. FOUND()
    
```

شكل ٤٨ - ٨ برنامج CUSTMT.PRG

```

32 | DO error WITH "Customer id. " + rtrim(m_custid) + " not found. "
33 | -----LOOP
34 | CASE FOUND()
35 |     * Get statement date and trial or final bill
36 |     m_ok = .T.
37 |     DO WHILE m_ok
38 |         m_date=IIF(DATE() > A->stmt_dat,DATE(),A->stmt_dat)
39 |         @ 16, 05 CLEAR TO 18,70
40 |         @ 16, 05 SAY " Enter statement date:" ;
40 |         GET m_date RANGE A->stmt_dat,
41 |         READ
42 |         pr = " "
43 |         DO WHILE .NOT. pr $ "YN"
44 |             @ 18, 05 SAY ;
44 |             " Date OK? (Y/N). If ok, setup printer and press Y" ;
44 |             GET pr PICTURE "I"
45 |             READ
46 |         ENDDO (While .NOT. pr $ "YN")
47 |         IF pr = "Y"
48 |             <-----EXIT
49 |             ENDIF
50 |         ENDDO (WHILE m_ok)
51 |         * Print statement for customer
52 |         @ 24, 30 SAY "printing statement..."
53 |         SET DEVICE TO PRINT
54 |         SET PRINT ON
55 |         SET CONSOLE OFF
56 |         DO custmtb
57 |         * Reset Printer & Console
58 |         SET DEVICE TO SCREEN
59 |         SET CONSOLE ON
60 |         SET PRINT OFF
61 |     ENDCASE
62 | ENDDO (.T.)
63 | CLOSE DATABASES
64 | RETURN
65 |
66 | *****
67 | * End of file CUSTMT.PRG *
68 | *****

```

تابع شكل ٤٨ - ٨ برنامج CUSTMT.PRG

- يفتح البرنامج دوائر صغيرة تتكرر ما لم يدخل المستخدم حرفي Y أو N لتأكيد أن التاريخ سليم والهدف من الدوائر ألا يقبل البرنامج غير حرفي Y و N (سطر ٤٣ - ٤٦).

- بعد أن يتأكد البرنامج من صحة التاريخ الذي أدخل يخرج من الدائرة الداخلية (سطر ٤٧ - ٤٩) ويجهز الطابعة لطباعة التقرير ويوقف عمل الشاشة أثناء طباعة التقرير ثم يستدعي البرنامج الذي سيتولى الطباعة وهو برنامج CUSTMTB.PRG

وبعد انتهاء الطباعة تعود كل من الطابعة والشاشة إلى وضعهما الطبيعي قبل بدء الطباعة (سطر ٥٣ - ٦٠) وستتناول بالشرح طريقة عمل برنامج CUSTMTB.PRГ الذي يتولى طباعة كشف الحساب.

برنامج CUSTMTB.PRГ

يشتمل شكل ٤٩ - ٨ على محتويات برنامج CUSTMTB.PRГ وعن هذا البرنامج نوضح ما يلي:

```

1 * -----*
2 * Program      : CUSTMTB.PRГ          *
3 * Author       : Magdi M. Abu Al-Ata *
4 * Date        : October 24, 1990     *
5 * Purpose      : Print satement for currently selected customer. *
6 * Called from: CUSTMT.PRГ            *
7 * Called from: CUSREP.PRГ            *
8 * Called from: CUSMENU.PRГ           *
9 * Called from: DPAPP.PRГ             *
10 * -----*
11 PRIVATE lines,m_debit,m_credt,mbal,hd
12 * Print titles and headings
13 hd= "Statement of account"
14 @ 1, 2 SAY mdate
15 @ 1,65 SAY "Time: "+TIME()
16 @ 3,80/2-(LEN(systitle)/2) SAY systitle
17 @ 4,80/2-(LEN(title2)/2) SAY title2
18 @ 5,80/2-(LEN(hd)/2) SAY hd
19 @ 6,80/2-(LEN(hd)/2) SAY REPLICATE(CHR(45),LEN(hd))
20 @ 9, 1 SAY "Account ID: "+A->cust_id
21 @ 10, 1 SAY REPLICATE("-",16)
22 @ 12, 5 SAY TRIM(A->fnam)+" " + TRIM(A->mmam)+" " +A->lnam
23 @ 13, 5 SAY trim(A->company)+" Company,"
24 @ 14, 5 SAY TRIM(A->addr)+" "
25 @ 15, 5 SAY TRIM(A->city)+" "
26
27 * Print statement column headings
28 @ 17, 1 SAY REPLICATE(CHR(45),78)
29 @ 18, 2 SAY ;
30 "Date          Description          Debit          Credit          Balance"
31 @ 19, 1 SAY REPLICATE(CHR(45),78)
32 lines = 20
33
33 * Print balance from last statement
34 IF DTOC(A->stmt_dat) # ' '

```

تابع شكل ٤٩ - ٨ برنامج CUSTMT.PRГ

القبول الثاني: نظام حسابات العملاء

```

35 | @ lines, 2 SAY DTOC(A->stmt_dat)+"          "***previous balance***"
36 | @ lines,67 SAY A->stmt_bal PICTURE "9,999.99"
37 | lines=lines+1
38 ENDIF
39
40 * Select transaction file, all unposted records
41 * before selected date of statement for selected customer
42 SELECT cuhist
43 SET FILTER TO cust_id=m_custid .AND. .NOT. posted .AND. date <= m_date
44 GO TOP
45
46 * Loop to print all unposted transactions thru specified date for account
47 m_debit=0          && Initialize total debits
48 m_credit=0         && Initialize total credits
49 mbal=A->stmt_bal   && Initialize running balance
50 DO WHILE .not.EOF() && Loop for each transaction.
51 | * Print transaction records
52 |   m_debit=m_debit+debit
53 |   m_credit=m_credit+credit
54 |   mbal=mbal+debit-credit
55 |   @ lines, 2 SAY date
56 |   @ lines,15 SAY desc
57 |   @ lines, 40 SAY debit PICTURE "@Z 9,999.99"
58 |   @ lines, 53 SAY - credit PICTURE "@Z 9,999.99"
59 |   @ lines, 67 SAY mbal PICTURE "9,999.99"
60 |   lines=lines+1
61 |   * skip to next record.
62 |   SKIP
63 ENDDO
64
65 * Print account summary
66 lines = lines + 4
67 @ lines, 1 SAY "ACCOUNT SUMMARY"
68 lines = lines+1
69 @ lines, 1 SAY "*****"
70 lines = lines+2
71 IF DTOC(A->stmt_dat) # ' '          && If prev. balance
72 |   @ lines,9 SAY "Previous balance"
73 |   @ lines,67 SAY A->stmt_bal PICTURE "9,999.99"
74 |   lines=lines+1
75 ENDIF
76 @ lines, 9 SAY "Payments received"
77 @ lines, 66 SAY - m_credit PICTURE "9,999.99"
78 lines = lines + 1
79 @ lines, 9 SAY "Current month charges"
80 @ lines, 66 SAY m_debit PICTURE "9,999.99"
81 @ lines + 1,66 SAY REPLICATE("-",08)
82 @ lines + 2, 9 SAY "CURRENT BALANCE"
83 @ lines + 2,66 SAY A->stmt_bal+m_debit-m_credit PICTURE "9,999.99"
84 @ lines + 3, 9 SAY "*****"
85 IF (A->stmt_bal+m_debit-m_credit) > 0          && If current balance:
86 |   @ lines + 6,24 SAY "Please pay your balance before " +DTOC(m_date+30)
87 |   @ lines + 7,24 SAY "*****"
88 ENDIF

```

تابع شكل ٤٩ - ٨

```

89
90 EJECT
91
92 * Update customer statement balance and date
93 * Update status of all specified transactions to posted
94 SELECT cuhist          &&Filtered file contains only unposted
95 GO TOP                 &&transactions shown on this statement.
96 REPLACE ALL posted WITH .T.
97
98 * Update statement date and balance in customer file
99 SELECT custmr
100 REPLACE stmt_bal WITH stmt_bal+m_debit-m_credt
101 REPLACE stmt_dat with m_date
102
103 SELECT cuhist
104 SET FILTER TO          && Reset filter on transaction file
105 RETURN
106
107 *****
108 * End of file CUSTMTB.PRG *
109 *****

```

تابع شكل ٤٩ - ٨

- يبدأ البرنامج بطباعة اسم الشركة وعنوانها واسم التقرير كبقية برامج التقارير (سطر ١٣ - ١٩).
- يطبع رقم الحساب واسم العميل وعنوانه في الأماكن المناسبة (سطر ٢٠ - ٢٥).
- يطبع عناوين أعمدة التقرير حسب المساحة المتاحة من الورق ويشمل التقرير الأعمدة التالية:
- التاريخ - البيان - العملية المدينة - العملية الدائنة - الرصيد (سطر ٢٨ - ٣٠).
- يطبع الرصيد الموجود بآخر كشف حساب ويطبع تاريخ آخر كشف حساب تحت التاريخ متبوعاً بعبارة: **Previous balance** في أول سطر من الكشف (سطر ٣٤ - ٣٨).
- يختار ملف المعاملات التاريخي ويختار منه فقط السجلات التي تتوافر فيها ثلاثة شروط هي:
- الأول: تشتمل على نفس رقم الحساب المختار.
- الثاني: تشتمل على القيمة F. في حقل Posted (أي لم ترسل في كشف سابق).
- الثالث: تاريخها أقل من أو يساوي التاريخ المختار.

- ويوضع المؤشر عند أول سجل بعد الاختيار (سطر ٤٢ - ٤٤).
- ينشئ عدادات لإجمالي العمليات المدينة وإجمالي العمليات الدائنة والرصيد (سطر ٤٧ - ٤٩).
- ينشئ دواة لطباعة سجلات الملف التاريخي التي تشتمل على رقم الحساب المختار يتم الخروج منها إذا انتهت سجلات الملف (سطر ٥٠ - ٦٣) وفي داخل هذه الدواة تزداد عدادات إجمالي المديونيات وإجمالي المدفوعات ويتعدل الرصيد حسب نوع العملية.
- بعد الانتهاء من طباعة السجلات المختارة يطبع ملخص للحساب يشمل البيانات التالية:

- أ - الرصيد السابق ويؤخذ من رصيد كشف الحساب السابق.
 - ب - إجمالي المدفوعات خلال الشهر. ويؤخذ من العداد الذي أنشأناه.
 - ج - إجمالي استحقاقات هذا الشهر. ويؤخذ من العداد الذي أنشأناه.
 - د - الرصيد الحالي وبحسب هكذا:
- الرصيد السابق + إجمالي الاستحقاقات - إجمالي المدفوعات (سطر ٧١ - ٨٤)

- وتختار المواقع المناسبة لكل هذه البيانات.
- إذا زاد الرصيد الحالي عن صفر يطبع في نهاية التقرير رسالة للعميل تطلب منه السداد خلال ثلاثين يومًا من تاريخ كشف الحساب (سطر ٨٥ - ٨٨).
 - بعد الانتهاء من طباعة كشف الحساب يتم تعديل حقل POSTED في الملف التاريخي وذلك باستبدال القيمة F. بالقيمة T. ليفهم البرنامج في الشهر القادم أن هذه العمليات اختيرت في كشف الحساب السابق (سطر ٩٤ - ٩٦) ولا يختارها مرة ثانية.

ملاحظة: يشتمل شكل ٥٠ - ٨ على كشف الحساب رقم D500 بعد طباعة الكشف الشهري الموجود في شكل ٤٦ - ٨ مباشرة ومنه تلاحظ أنه يشتمل على الرصيد السابق وأن الرصيد السابق هو نفسه الرصيد الحالي. ولا يشتمل على حركات مدينة

Friday 1, March 1991		Time: 09:49:25		
Arabian Computer Center C.R. 39387 Statement of account -----				
Account ID: D500				
Adel Arafa Bahi Al-Din Computech Inc. Company, Al manar building, Cairo.				
Date	Description	Debit	Credit	Balance
01/03/91	**Previous balance**			1,650.00
ACCOUNT SUMMARY				

	Previous balance			1,650.00
	Payments received		0.00	
	Current month charges		0.00	
	CURRENT BALANCE			1,650.00

Please pay your balance before 31/03/91 *****				

شكل ٥٠ - كشف حساب لا يشتمل على معاملات جديدة

الفصل الثامن : نظام حسابات العملاء

أو دائنة لأنها اختبرت كلها في الكشف السابق فإذا أضيفت حركات جديدة فستظهر عند طباعة الكشف التالي وسيعدل الرصيد أيضًا تبعًا لها .

- ثم يتم تعديل حقيقي تاريخ كشف الحساب والرصيد بملف العملاء الرئيسي لهذا الحساب بالتاريخ والرصيد الحاليين (سطر ٩٩ - ١٠١) .
- يعاد الملف التاريخي إلى وضعه الطبيعي . أي يبدو مشتملاً على كل سجلاته .

الفصل التاسع

صيانة الملفات

لا يخلو أى نظام لإدارة قواعد البيانات من مجموعة من البرامج تقوم بصيانة الملفات. مثل عمل نسخ احتياطية للملفات تحسباً للتلف أو الضياع الذى يحدث في حالات كثيرة. أو استرجاع النسخ الاحتياطية في حالة الحاجة إليها من الأقراص اللينة أو هدف السجلات غير الضرورية من الملفات أو عمل الفهارس اللازمة للملفات.

وسوف نناقش في هذا الفصل هذه الوظائف وغيرها ويمكنك فصل هذه الوظائف ووضعها مع أى نظام من الأنظمة السابقة في حالة تشغيله منفرداً أو إبقائها على حالها في حالة ربط الأنظمة كلها من خلال قائمة واحدة.

سنشرح وفي هذا الفصل كيفية حفظ واسترجاع الملفات بالإمكانات المتاحة في قاعدة البيانات. إلا أن هذا الأسلوب يعاب عليه بطء التنفيذ ولذلك فهو يناسب الأعمال الصغيرة أما في الأعمال الكبيرة التي تحتاج لحفظ ملفات كبيرة واسترجاعها فننصح بشراء وحدة شرائط خاصة لهذا الغرض ذات كفاءة عالية وهي قليلة المشاكل وتشتمل على برامج خاصة لأغراض الحفظ والاسترجاع. وتسمى هذه الوحدة:

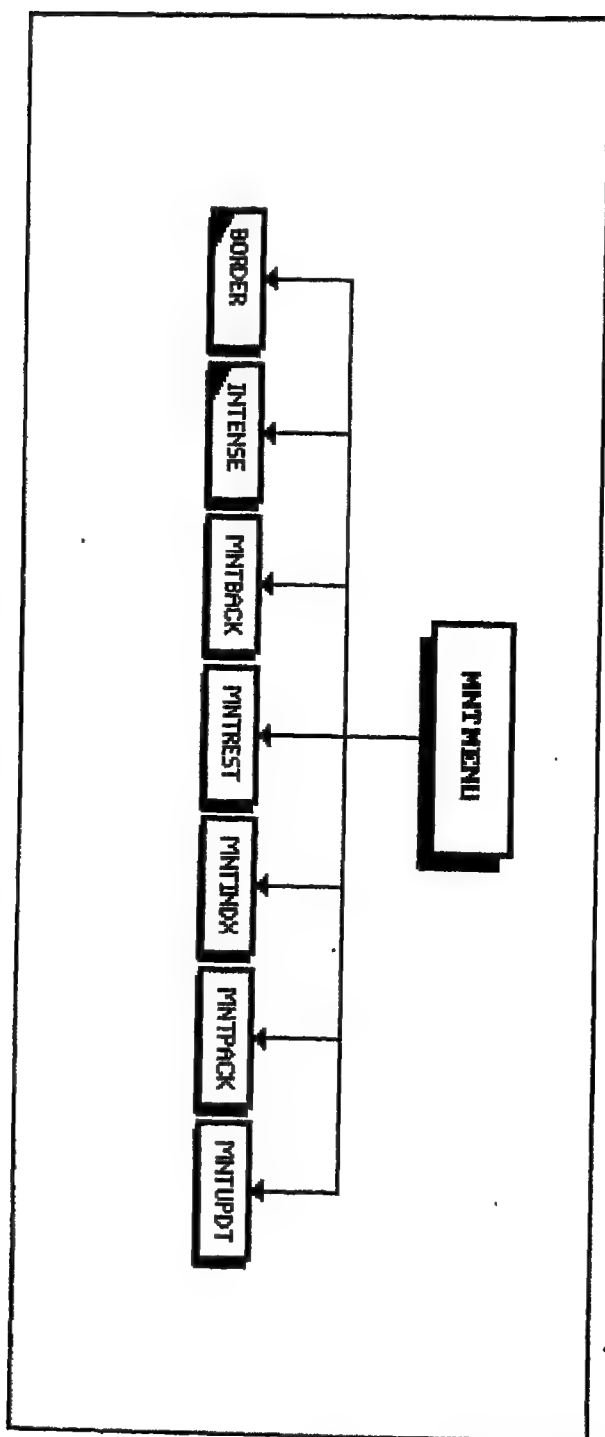
. Backup tape unit

بالإضافة إلى وظائف أخرى لا تستغني عنها نظم إدارة قواعد البيانات مثل حذف السجلات غير الضرورية أو عمل الفهارس أو تحديث الملفات.

والنظام الذي سنشرحه في هذا الفصل لا يشتمل على ملفات ولا ينشئ ملفات جديدة لأنه يقوم بخدمة وصيانة الملفات التي أنشأناها في الفصول السابقة ويشتمل على خمسة برامج تقوم بوظائف صيانة الملفات. ويمكن أن يضاف إلى هذه البرامج أي برامج عامة أخرى تراها ضرورية. ويشتمل هذا النظام على الوظائف التالية:

- ١ - عمل نسخ احتياطية للملفات على أقراص لينة.
- ٢ - استرجاع النسخ الاحتياطية من الأقراص اللينة.
- ٣ - فهرسة أو إعادة فهرسة الملفات.
- ٤ - تنظيف الملفات من السجلات الغير ضرورية.
- ٥ - تعديل أرصدة المخازن بعد عمليات البيع والشراء.

ويوضح شكل ١ - ٩ خريطة النظام التي توضح العلاقة بين البرامج التي يشتمل عليها ومن هذا الشكل تلاحظ أن البرنامج الرئيسي في النظام وهو MNTMENU.PRG يستدعي خمسة برامج أخرى كل منها يقوم بوظيفة محدودة ويمثل كل برنامج من هذه البرامج مستطيل داخل خريطة النظام بالإضافة إلى ١ راءين سبق شرحهما تعرفهما بتظليل الركن اليسار السفلي من المستطيل. وستتناول فيما يلي شرح برامج النظام والأفكار الجديدة التي يشتمل عليها.



شكل ٩ - ١ خريطة نظام صيانة الملفات

قائمة صيانة الملفات

يشتمل شكل ٢ - ٩ على قائمة صيانة الملفات والاختيارات التي تشتمل عليها ويقوم كل اختيار باستدعاء وظيفة من الوظائف التي أشرنا إليها. كما يشتمل شكل

Tuesday 26, February 1991
Time: 15:30:17

Arabian Computer Center
C.R. 39387

Maintenance menu

- (1) Back up to floppy diskette(s) ..
- (2) Restore from floppy diskette(s)
- (3) Index/Reindex
- (4) Pack data files
- (5) Update inventory
- (6) Go to main menu
- (Q) Quit to DOS

Select number or colored letter ==> █

شكل ٢ - ٩ قائمة صيانة الملفات

٣ - ٩ على البرنامج اللازم لإظهار هذه القائمة وتنفيذ اختياراتها وقد سبق أن شرحنا فكرة برامج القوائم في الفصول السابقة وهذا البرنامج لا يختلف عنها في شيء. ونوضح فيما يلي البرامج التي تقوم بصيانة الملفات.

عمل نسخ احتياطية للملفات

البرنامج اللازم لعمل نسخ احتياطية من الملفات على أقراص لينة يمكن استرجاعها إذا دعت ضرورة إلى ذلك هو MNTBACK.PRГ.

```

1 * -----*
2 * Program.....: MNTMENU.PRG *
3 * Called form.: DBAPP.PRG *
4 * Programmer...: Magdi M. Abu Al-Ata *
5 * Date.....: Dec. 25,1990 *
6 * Purpose.....: To display maintenace menu *
7 * -----*
8 PRIVATE scrttitle,scrfootr,choice
9 scrttitle = "Maintenance menu"
10 scrfootr = "Select number or colored letter ==>"
11 ok = .T.
12 DO WHILE ok
13     choice = " "
14     DO border WITH scrttitle,scrfootr
15     * Draw menu
16     @10, 5 SAY "(1) Back up to floppy diskette(s).."
17     DO intense WITH 10,10,"B"
18     @11, 5 SAY "(2) Restore from floppy disketee(s)"
19     DO intense WITH 11,10,"R"
20     @12, 5 SAY "(3) Index/Reindex....."
21     DO intense WITH 12,10,"I"
22     @13, 5 SAY "(4) Pack data files....."
23     DO intense WITH 13,10,"p"
24     @14, 5 SAY "(5) Update inventory ....."
25     DO intense WITH 14,10,"u"
26     @15, 5 SAY "(G) Go to main menu....."
27     DO intense WITH 15,10,"G"
28     @16, 5 SAY "(Q) Quit to DOS....."
29     DO intense WITH 16,10,"q"
30     @22,43 GET choice
31     READ
32     * Do selected option
33     DO CASE
34     CASE choice $ "1Bb"
35         DO mntback                                && Back up files
36     CASE choice $ "2rR"
37         DO mntrest                                && Restore files
38     CASE choice $ "3Ii"
39         DO mtindx                                 && Index/Reindex
40     CASE choice $ "4pP"
41         DO mntpack                                && Remove unwanted records
42     CASE choice $ "5Uu"
43         DO mntupdt                                && Adjust inventory
44     CASE choice $ "Gg"
45         RETURN                                    && Go to Main menu
46     CASE choice $ "qQ"
47         CLEAR
48     <=====QUIT                                && Quit to DOS
49     OTHERWISE
50     <-----LOOP
51     ENDCASE
52 ENDD
53 *****
54 * End of file MNTMENU.PRG *
55 *****

```

شكل ٣ - ٩ برنامج MNTMENU.PRG

ويشتمل شكل ٤ - ٩ على هذا البرنامج. ونوضح فيما يلي فكرة البرنامج وخطوات تنفيذه.

لأن البرنامج سيستغرق بعض الوقت فمن المناسب أن تظهر رسالة للمستفيد تخبره بذلك حتى لا يشك أن الحاسب تعطل أو حصل شيء غير عادي. ولذلك فإننا نظهر هذه الرسالة في البداية (سطر ١٢) ونعطي المستفيد الفرصة ليختار الاستمرار

```

1 * ----- *
2 * Program.....: MNTBACK.PRG *
3 * Called from.: MNTMENU.PRG *
4 * Called from.: DBAPP.PRG *
5 * Programmer...: Magdi M. Abu Al-Ata *
6 * Date.....: Dec. 25,1990 *
7 * Purpose.....: Back up data files to floppy diskettes *
8 * ----- *
9 PRIVATE yn
10 yn = " "
11 CLEAR
12 @ 1,2 SAY "This option will take betwen 1 and 20 minutes"
13 @ 2,2 SAY "Do you have formated diskette(s) for back up? [Y/N]" GET yn
14 READ
15 IF UPPER(yn) = "Y"
16     source = "C:"
17     target = "A:"                && Target drive for backup
18     files1 = source+"ORDER.*"    && To include .DBT & .DBF
19     files2 = source+"INV.*"
20     files3 = source+"PURCH.*"
21     files4 = source+"SALE.*"
22     files5 = source+"CUSTMR.*"
23     files6 = source+"CUHIST.*"
24     RUN BACKUP &files1 &target
25     RUN BACKUP &files2 &target/A    && /A Append those files to existing
26                                     && files on the same drive
27     RUN BACKUP &files3 &target/A
28     RUN BACKUP &files4 &target/A
29     RUN BACKUP &files5 &target/A
30     RUN BACKUP &files6 &target/A
31     ??CHR(7)
32     CLEAR
33     WAIT ;
34     "End backing up ...Remove your diskettes and press any key to continue"
35 ENDIF (yn = "Y")
36 RELEASE yn
37 RETURN
38
39 *****
40 * End of file MNTBACK.PRG *
41 *****

```

شكل ٤ - ٩ برنامج MNTBACK.PRG

في التنفيذ أو الرجوع عنه . (سطر ١٣ - ١٤) . فإذا قرر الاستمرار في التنفيذ يتم الآتي :

- تخزين اسم الوحدة التي تشتمل على الملفات واسم الوحدة التي سننسخ إليها بحقول الذاكرة (سطر ١٦ - ١٧) .

- تخزين أسماء الملفات بحقول ذاكرة (سطر ١٨ - ٢٣) . وتلاحظ أننا نختار مع اسم الملف كل أسماء الامتداد التي تخصه لكي يتم نسخ الملفات من نوع DBT . إن وجدت .

- استخدمنا أمر RUN (سطر ٢٤) لأول مرة ومعناه تنفيذ الخروج من قاعدة البيانات وتنفيذ أحد برامج نظام التشغيل (DOS) والعودة للبرنامج بعد الانتهاء من تنفيذ البرنامج الخارجي . ويجب الانتباه إلى أن أمر RUN يحتاج لمساحة قدرها ١٧ ك . ب بالإضافة إلى المساحة التي يشغلها برنامج «الدوس» . ولذلك يجب أن نتأكد أن مساحة الذاكرة عندك تكفي كل ذلك . وقد استخدمنا أمر BAKCUP بنفس الطريقة التي يستخدم بها مع نظام التشغيل . ومعنى الأمر الموجود في سطر رقم ٢٤ كما يلي : مطلوب عمل نسخ احتياطية (BACKUP) للملفات المشار إليها بالمتغير files1 (وهي *C:\CUSTMR) إلى القرص الموجود في الوحدة المشار إليها بالمتغير target (وهي A:) .

استخدمنا الوظيفة & وهي تستخدم للإشارة إلى محتويات حقل الذاكرة .
- علامة A/ في سطر ٢٥ - ٢٩ تعني نسخ الملفات المشار إليها بعد آخر ملف موجود على القرص .
- بعد الانتهاء من حفظ الملفات تظهر رسالة للمستفيد لتطمئنه أن العمل انتهى (سطر ٣٢) .

استرجاع الملفات

يشتمل شكل ٥ - ٩ على برنامج MNTREST.PRГ ومهمته (استرجاع الملفات المحفوظة على أقراص لينة باستخدام أمر BACKUP .

```

1 * ----- *
2 * Program.....: MNTREST.PRG *
3 * Called from.: MNTMENU.PRG *
4 * Called from.: DBAPP.PRG *
5 * Programmer..: Magdi M. Abu Al-Ata *
6 * Date.....: Dec. 25,1990 *
7 * Purpose.....: Restore the data files from floppy diskette(s) *
8 * ----- *
9 PRIVATE yesno
10 yesno = " "
11 CLEAR
12 @ 1,2 SAY "This option will take between 1 and 20 minutes"
13 @ 2,2 SAY "Do you have your diskette(s) for restore? [Y/N]" GET yesno
14 READ
15 IF UPPER(yesno) = "Y"
16     source = "A:"                && Source drive for restore
17     target = "C:"                && Target drive to accept files
18     files1 = target+"*.*"        && All files in drive A: will be
19                                   && .DBF, .DBT and .NDX
20     RUN RESTORE &source &files1
21 ENDIF (yesno = "Y")
22 RELEASE yesno
23 RETURN
24
25 *****
26 * End of file MNTREST.PRG *
27 *****

```

شكل ٥ .. ٩ برنامج MNTREST.PRG

والبرنامج يستخدم نفس فكرة برنامج الحفظ ويستخدم أمر RESTORE لهذا الغرض.

فهرسة الملفات وإعادة فهرستها

أوضحنا في الفصول السابقة أن ملفات الفهرسة تنشأ بعد إنشاء ملفات قاعدة البيانات. واستخدمنا هذا الأسلوب في الفصول السابقة. وقد أوردنا هنا برنامجاً خاصاً لهذا الغرض لسببين: السبب الأول أنك إذا كنت تنوي تطوير برامج وبيعها في الأسواق فمن المناسب أن تتبع هذا الأسلوب الذي يعتمد على إنشاء الملفات تلقائياً عند تركيب النظام لدى العميل والسبب الثاني أننا نحتاج لإعادة فهرسة الملفات لأسباب كثيرة ويتولى هذا البرنامج أيضاً هذه العملية.

ويشتمل شكل ٦ - ٩ على برنامج MNTINDX.PRG اللازم لهذا الغرض. والبرنامج يبدأ بإظهار رسالة توضح للمستفيد أن هذا العمل سيستغرق بعض الوقت وعليه أن يقرر الاستمرار في التنفيذ أو الرجوع عنه وذلك لأن كلا من أمر

```

1 * ----- *
2 * Program.....: MNTINDX.PRG *
3 * Called from.: MNTMENU.PRG *
4 * Called from.: DBAPP.PRG *
5 * Programmer...: Magdi M. Abu Al-Ata *
6 * Date.....: Dec. 25,1990 *
7 * Purpose.....: Rebuild the index files *
8 * ----- *
9 PRIVATE yes
10 yes = " "
11 @24,1 SAY "This procedure will run slowly. Continue? (Y/N) " ;
11 GET yes PICTURE "I"
12 READ
13 IF yes = "Y"
14     USE custmr
15     @24,0
16     @24,20 SAY "Please wait...Rebuilding CUSTMR.NDX"
17     IF .NOT. FILE("custmr.ndx")
18         INDEX ON cust_id TO custmr
19     ELSE
20         SET INDEX TO custmr
21         REINDEX
22     ENDIF
23     @24,20 SAY "Please wait...Rebuilding NAME.NDX"
24     IF .NOT. FILE("name.ndx")
25         INDEX ON fnam+mmam+lnam TO name
26     ELSE
27         SET INDEX TO name
28         REINDEX
29     ENDIF
30     *
31     USE cuhist
32     @24,0
33     @24,20 SAY "Please wait...Rebuilding CUHIST.NDX"
34     IF .NOT. FILE("cuhist.ndx")
35         INDEX ON cust_id+STR(YEAR(date),4)+ STR(MONTH(date),2)+ ;
35             STR(DAY(date),2) TO cuhist
36     ELSE
37         SET INDEX TO cuhist
38         REINDEX
39     ENDIF
40     *
41     USE inv
42     @24,0
43     @24,20 SAY "Please wait...Rebuilding ITEM.NDX"
44     IF .NOT. FILE("item.ndx")

```

شكل ٦ - ٩ برنامج MNTINDX.PRG

```

45 | INDEX ON item_no TO item
46 | ELSE
47 | SET INDEX TO item
48 | REINDEX
49 | ENDIF
50 | @24,0
51 | @24,20 SAY "Please wait...Rebuilding DESC.NDX"
52 | IF .NOT. FILE("desc.ndx")
53 | INDEX ON desc TO desc
54 | ELSE
55 | SET INDEX TO desc
56 | REINDEX
57 | ENDIF
58 | *
59 | USE order
60 | @ 24,0
61 | @24,20 SAY "Please wait...Rebuilding ORDER.NDX"
62 | IF .NOT. FILE("order.ndx")
63 | INDEX ON ord_no TO order
64 | ELSE
65 | SET INDEX TO order
66 | REINDEX
67 | ENDIF
68 | @ 24,0
69 | @24,20 SAY "Please wait...Rebuilding IDATE.NDX"
70 | IF .NOT. FILE("idate.ndx")
71 | INDEX ON ord_dat TO idate
72 | ELSE
73 | SET INDEX TO idate
74 | REINDEX
75 | ENDIF
76 | *
77 | USE purch
78 | @ 24,0
79 | @24,20 SAY "Please wait...Rebuilding PUR_INV.NDX"
80 | IF .NOT. FILE("pur_inv.ndx")
81 | INDEX ON invoice_no TO pur_inv
82 | ELSE
83 | SET INDEX TO pur_inv
84 | REINDEX
85 | ENDIF
86 | @ 24,0
87 | @24,20 SAY "Please wait...Rebuilding PUR_ORD.NDX"
88 | IF .NOT. FILE("pur_ord.ndx")
89 | INDEX ON ord_no TO pur_ord
90 | ELSE
91 | SET INDEX TO pur_ord
92 | REINDEX
93 | ENDIF
94 | @ 24,0
95 | @24,20 SAY "Please wait...Rebuilding ORDINVU.NDX"
96 | IF .NOT. FILE("ordinvu.ndx")
97 | INDEX ON ord_no+invoice_no TO ordinvu UNIQUE
98 | ELSE

```

```

99      SET INDEX TO ordinvu
100     REINDEX
101     ENDIF
102     @ 24,0
103     @24,20 SAY "Please wait...Rebuilding ORDINV.NDX"
104     IF .NOT. FILE("ordinv.ndx")
105     |     INDEX ON ord_no+invoice_no TO ordinvu
106     | ELSE
107     |     SET INDEX TO ordinv
108     |     REINDEX
109     | ENDIF
110     @ 24,0
111     @24,20 SAY "Please wait...Rebuilding ITEM_P.NDX"
112     IF .NOT. FILE("item_p.ndx")
113     |     INDEX ON item_no TO item_p
114     | ELSE
115     |     SET INDEX TO item_p
116     |     REINDEX
117     | ENDIF
118     *
119     USE sale
120     @ 24,0
121     @24,20 SAY "Please wait...Rebuilding SAL_INV.NDX"
122     IF .NOT. FILE("sal_inv.ndx")
123     |     INDEX ON inv_no TO sal_inv
124     | ELSE
125     |     SET INDEX TO sal_inv
126     |     REINDEX
127     | ENDIF
128     @ 24,0
129     @24,20 SAY "Please wait...Rebuilding ITEM_S.NDX"
130     IF .NOT. FILE("item_s.ndx")
131     |     INDEX ON item_no TO item_s
132     | ELSE
133     |     SET INDEX TO item_s
134     |     REINDEX
135     | ENDIF
136 ENDIF
137 RETURN
138
139 *****
140 * End of file MNTINDX.PRG *
141 *****

```

تابع شكل ٦ - ٩ برنامج MNTINDX.PRG

REINDEX أو INDEX ينفذ ببطء إلى حد ما خصوصاً إذا كانت الملفات كبيرة. فإذا قرر المستخدم الاستمرار في التنفيذ فإن البرنامج يبدأ في فتح الملفات كلها واحداً تلو الآخر. وعندما يفتح أحد الملفات يتأكد من وجود ملف / ملفات الفهرسة المتصلة به فإذا وجدها أعاد فهرستها وإذا لم يجدها فهذا يعني أن البرنامج ينفذ لأول مرة وبالتالي قام بإنشائها.

- ونوضح فيما يلي الإجراء المتبع مع أحد الملفات وبنفس الطريقة تفهم باقي الملفات الموجودة بالبرنامج .
- تفتح ملف قاعدة البيانات ملف CUSTMR.DBF (سطر ١٤).
 - تظهر رسالة أن الملف يعاد فهرسته وتختفي عند انتهاء عملية الفهرسة أو إعادة الفهرسة (سطر ١٦).
 - يسأل البرنامج هل ملف CUSTMR.NDX موجود على نفس الدليل (سطر ١٧).
 - واستخدمنا الوظيفة (FILE) مع علامة .NOT. لهذا الغرض .
 - إذا لم يكن الملف موجوداً تم إنشاؤه بأمر INDEX المعروف (سطر ١٨). وهذا يحدث لمرة واحدة فقط وهي أول مرة ينفذ فيها البرنامج .
 - أما إذا كان موجوداً (سبق إنشاؤه أو سبق تنفيذ هذا البرنامج) فيتم اختيار الملف وإعادة فهرسته (سطر ٢٠ - ٢١).
 - يتم السؤال عن ملف الفهرس الثاني المتصل بنفس ملف قاعدة البيانات ونتبع معه نفس الإجراء السابق (سطر ٢٤ - ٢٩).
 - نختار ملف قاعدة البيانات التالي ونتعامل معه بنفس الأسلوب . . . وهكذا حتى تنتهي كل الملفات الموجودة بالنظام .

تنظيف الملفات من السجلات غير الضرورية

يشتمل شكل ٧ - ٩ على برنامج MNTPACK.PRГ . ومهمة هذا البرنامج مهمة بسيطة تتمثل في حذف السجلات المعلمة لغرض الحذف بأمر DELETE من الملفات حذفاً نهائياً وكما تذكر فقد اتبعنا في البرامج التي يشتمل عليها هذا الكتاب أسلوب تعليم السجلات للحذف فقط دون حذفها نهائياً . اعتماداً على أننا سنستخدم برنامجاً خاصاً يقوم بهذا الغرض عند انتهاء الدوام أو بعد وقت العمل لأن أمر PACK من الأوامر البطيئة نسبياً .

```

1 * ----- *
2 * Program.....: MNTPACK.PRG *
3 * Called from.: MNTMENU.PRG *
4 * Called from.: DBAPP.PRG *
5 * Programmer..: Magdi H. Abu Al-Ata *
6 * Date.....: Dec. 25,1990 *
7 * Purpose.....: Rebuild the index files" *
8 * Purpose.....: Remove records marked for deletion *
9 * ----- *
10 PRIVATE yes
11 yes = " "
12 @ 24,0
13 @24,2 SAY "Pcking records marked for deletion will take some time. " ;
13 + "Conyinue? (Y/N) " GET yes PICTURE "I"
14 READ
15 IF yes = "Y"
16     USE custmr
17     PACK
18     USE inv
19     PACK
20     USE order
21     PACK
22     USE purch
23     PACK
24     USE sale
25     PACK
26     USE cuhist
27     PACK
28 ENDIF
29 CLOSE DATABASES
30 RETURN
31
32 *****
33 * End of file MNTPACK.PRG *
34 *****

```

شكل ٧ - ٩ برنامج MNTPACK.PRG

تعديل أرصدة المخازن بعد عمليات الشراء والبيع

لابد من تعديل أرصدة المخازن بعد كل عملية شراء وذلك بإضافة الكميات المشتراة من صنف معين لنظيرتها في ملف المخازن وكذلك بعد كل عملية بيع لخصم الكميات المباعة من صنف معين من نظيرتها في ملف المخازن ويجوز أن تتم هذه العملية وتسمى تحديث الملف (update) بعد الشراء أو البيع مباشرة. كما يجوز أن تتم ببرنامج مستقل ينفذ عند نهاية اليوم أو الأسبوع أو في أي وقت تراه مناسباً. واختيار إحدى الطريقتين يرجع إلى سياسة المؤسسة.

وقد اخترنا أن تتم عملية تحديث الملفات برنامج مستقل لأن أمر UPDATE أيضاً بطيء نسبياً ولا نريد أن يتأثر العمل اليومي بعملية التحديث التي تتم بعد كل عملية بيع.

ويشتمل شكل ٨ - ٩ على برنامج MNTUPDT.PRG الذي يقوم بعملية التحديث.

```

1 * -----*
2 * Program.....: MNTUPDT.PRG*
3 * Called from.: MNTMENU.PRG*
4 * Called from.: DBAPP.PRG*
5 * Programmer...: Magdi M. Abu Al-Ata*
6 * Date.....: Dec. 25,1990*
7 * Purpose.....: Update master inventory file after purchase or sale*
8 * -----*
9 * The following segment updates INV.DBF file from PURCH.DBF
10 CLOSE DATABASES
11 SELECT A
12 USE INV
13 SELECT B
14 USE PURCH
15 SELECT C
16 USE SALE
17 * Filter PURCH.DBF to unupdated records
18 SELECT B
19 SET INDEX TO item_p
20 SET FILTER TO .NOT. updated      && Filter PURCH.DBF to new
21 .                                && purchased records
22 GO TOP                        && Put the pointer at the begining of filtered file
23 COUNT FOR .NOT. updated TO updp
24 IF updp > 0                    && If new record(s) added to PURCH.DBF
25     @ 24,0
26     @ 24,2 SAY "Updating master inventory file form purchase. Please wait"
27     SELECT A
28     SET INDEX TO item
29     * Update from PURCH.DBF
30     UPDATE ON item_no FROM PURCH REPLACE onstock WITH ;
30     onstock + B-> qty_rcv, updt_dat WITH date()
31     SELECT B
32     * Mark the updated records to exclude it next time
33     REPLACE ALL UPDATED WITH .T.
34 ENDIF
35 * Switch PURCH.DBF to its normal records
36 SET FILTER TO
37 @ 24,2 SAY "End of updating master file from purchase file"
38 *
39

```

```
40 * The following segment updates INV.DBF file from SALE.DBF
41 SELECT C                                && Sales file
42 SET INDEX TO item_s
43 SET FILTER TO .NOT. UPDATED
44 GO TOP
45 COUNT FOR .NOT. updated TO updsl
46 IF updsl > 0                            && Tf new record(s) added to PURCH.DBF
47   @ 24,0
48   @ 24,2 SAY "Updating master inventory file from sales. Please wait"
49   SELECT A
50   SET INDEX TO item                      && Indexed on item_no
51   * Update from SALE.DBF
52   UPDATE ON item_no FROM sale REPLACE onstock WITH ;
52   onstock - C-> qty_sold, updt_dat WITH date()
53   * Switch SALE.DBF to its normal records
54   SELECT C
55   * Mark the updated records to exclude it next time
56   REPLACE ALL UPDATED WITH .T.
57 ENDIF
58 * Switch SALE.DBF to its normal records
59 SET FILTER TO
60 @ 24,2 SAY "End of updating master file from sales file  "
61 CLOSE DATABASES
62 RETURN
63 *****
64 * End of file MNTUPDT.PRG *
65 *****
```

تابع شكل ٨ - ٩ برنامج MNTUPDT.PRG

والبرنامج يشتمل على جزئين. الجزء الأول خاص بتحديث ملف المخازن بعد الشراء والثاني خاص بتحديث ملف المخازن بعد البيع. وقد آثرت ذلك حتى إذا اخترت أن يتم التحديث في نهاية برامج الشراء أو البيع تستطيع نقل الجزء المناسب لعملك دون بذل جهد جديد.

- وفي هذا البرنامج يتم اختيار ملف المخزون الرئيسي في المنطقة A و ملف المشتريات في المنطقة B و ملف المبيعات في المنطقة C داخل ذاكرة الحاسب (سطر ١١ - ١٦).

- نختار ملف المشتريات ليكون هو الملف الحالي مرتباً حسب رقم الصنف (سطر ١٨ - ١٩).

- يتم اختيار السجلات التي لم تتعدل من قبل (سطر ٢٠) ويعرفها البرنامج بوجود علامة F. في حقل UPDATED ومعروف أن الحقول المنطقية تنشأ بقيمة F. ما لم تغيرها. ولأننا لم نستخدم هذا الحقل أثناء تسجيل المشتريات فسيظل مشتملاً على

١. لاحظ أن نفي الحقل المنطقي يتم باستخدام علامة .NOT.
 - يتم عد السجلات الموجودة في الملف بعد عملية الاختيار (التي لم تتعدل) (سطر ٢٣) فإذا لم توجد سجلات فإن الملف يرجع إلى حالته الطبيعية (سطر ٣٦) ويستبعد تنفيذ أمر .UPDATE.
 - إذا اشتمل الملف بعد الاختيار على سجلات لم تتعدل (تحدث) تظهر رسالة للمستفيد لأن عملية التحديث تستغرق بعض الوقت (سطر ٢٦) ويختار ملف المخازن مرتباً حسب رقم الصنف (سطر ٢٧ - ٢٨).
 - العمود الفقري في البرنامج هو أمر .UPDATE (في سطر ٣٠) وهو من الأوامر المعقدة نسبياً ولذلك نوضحه بالتفصيل كما يلي:
 - يقوم أمر .UPDATE بربط الملف الحالي مع ملف آخر موجود بمنطقة ثانوية طبقاً لمحتويات حقل مشترك بينهما ويسمى .key field . وهو في هذا المثال حقل item_no . ويجب أن يكون الملفين مفهرسين أو مفروزين لكي تتم عملية الربط وفي أمر .UPDATE الملف الحالي هو INV.DBF .
 - يتم ربط ملف PURCH الموجودة في المنطقة B طبقاً لمحتويات حقل item_no وهذا هو معنى الجزء الأول من الأمر
- UPDATE ON item_no FROM PURCH
- وأثناء عملية التحديث يتم استبدال محتويات حقلين . الحقل الأول onstock ويعني الكمية الموجودة ويتم استبداله بقيمة جديدة هي الكمية الموجودة .(onstock) + الكمية المشتراة (B->qty_rcv) . وقد استخدمنا العلامة B-> للإشارة إلى أن حقل qty_rcv موجود بالمنطقة الثانوية B .
- والحقل الثاني هو حقل updt_dat ويتم استبداله بالتاريخ المسجل في الحاسب اليوم .
- بعد انتهاء عملية التحديث نختار ملف المشتريات (سطر ٣١) ويتم استبدال علامة F. بعلامة T. الموجودة بحقل updated في كل السجلات (سطر ٣٣) .
- سواء نفذنا أمر UPDATE أم لا يجب إرجاع ملف المشتريات إلى حالته السابقة (سطر ٣٦) .

الجزء الثاني من البرنامج يستخدم نفس المفهوم والاختلاف الوحيد في استخدام أمر UPDATE بدلاً من إضافة الكمية المشتراة إلى كمية المخزون مطرح الكمية المباعة من كمية المخزون.

الفصل العاشر

تعريب النظام

لأن قاعدة البيانات كتبت باللغة الانجليزية فقد كتبنا جميع البرامج باللغة الانجليزية فإذا أردت تعريب النظم التي أوردناها في هذا الكتاب فلا بد من اللجوء إلى أحد برامج تعريب مدخلات ومخرجات الحاسب المتاحة في السوق مثل مساعد العربي/٢ أو نافذة وسنشرح في هذا الفصل كيفية استخدام كل منهما لتعريب النظام بافتراض أنهما من أشهر البرامج الموجودة الآن في الأسواق لتعريب مدخلات ومخرجات الحاسب.

تعريب النظام باستخدام مساعد العربي / ٢

برنامج مساعد العربي / ٢ من إنتاج شركة سعودي سوفت بالمملكة العربية السعودية وهو من البرامج التي تتيح تعريب مدخلات ومخرجات الحاسب. وتعتمد فكرته على تحميل البرنامج في ذاكرة الحاسب. ويبقى في الذاكرة حتى يتم إغلاق الحاسب أو إصدار أمر مناسب لاستبعاده.

تشغيل مساعد العربي

قبل تشغيل مساعد العربي يجب تركيبه على قرص مرن أو صلب (راجع الفصل الرابع من كتيب Installation Guide إذا لم تكن تعرف طريقة تركيب مساعد العربي). ويفرض أن مساعد العربي تم تركيبه أدخل الأمر الآتي من محث نظام التشغيل

C:\<MA

إذا كان التشغيل سيتم من قرص صلب (بعد الانتقال إلى دليل مساعد العربي أو تعديل ملف Autoexec.bat ليشمل تشغيل مساعد العربي من أي دليل).

أما إذا أردت تشغيل المساعد ومركز الإشراف (MACC) * معاً فيجب أن تدخل الأمر بهذا الشكل.

C:\<MA/S

فإذا حملت مساعد العربي / ٢ وأردت بعد ذلك تحميل مركز الإشراف (MACC) أدخل هذا الأمر:

C:\<MACC

* مركز الإشراف عبارة عن قائمة أفقية ذات قوائم منسدلة تعطيك معلومات عن الإمكانيات المتاحة بالبرنامج ويمكنك من تغيير كل أو بعض هذه الإمكانيات.

اختيار اتجاه الكتابة على الشاشة واللغة

هناك حالتان لاتجاه الكتابة على الشاشة الأولى: اتجاه الكتابة من اليسار إلى اليمين وهو الوضع الأصلي للكتابة بعد تحميل مساعد العربي / ٢ مباشرة، والثانية اتجاه الكتابة من اليمين إلى اليسار وهو الاتجاه الذي يتيح مساعد العربي / ٢ كذلك يمكن الكتابة بالحروف العربية كما يمكن الكتابة بالحروف اللاتينية والكتابة بالحروف اللاتينية هي الوضع الأصلي للكتابة بعد تحميل مساعد العربي / ٢. وتسهيلاً للاستخدام فقد وضع مصمموا مساعد العربي / ٢ في حسابهم أن يتم تغيير اتجاه الكتابة على الشاشة ولغة الكتابة معاً. ويتم تغيير اتجاه الكتابة واللغة معاً من حالة إلى أخرى بضغط مفتاحي Alt + NumLock بيد أنه بإمكانك تغيير اللغة من داخل الاتجاه وذلك بضغط مفتاحي Alt + Right Shift للكتابة العربي إذا كان اتجاه الشاشة من اليسار إلى اليمين أو مفتاحي Alt + Left Shift إذا كان اتجاه الشاشة من اليمين إلى اليسار.

ويمكن الاستفادة من إمكانية تغيير اللغة واتجاه الشاشة لإعادة كتابة البيانات النصية الموجودة في البرامج أو لإدخال بيانات الملفات إذا كانت بيانات الملفات عربية.

استخدم منسق النصوص أو محرر السطور المؤلف لك لإعادة كتابة البرامج لتغيير النصوص والرسائل الموجودة بالبرنامج إلى نصوص ورسائل باللغة العربية.

ويمكن تشغيل النظام بالإمكانية التي شرحناها وهي تغيير اتجاه الشاشة واللغة بواسطة لوحة المفاتيح إلا أننا لا ننصح بهذا الأسلوب خصوصاً بالنسبة للنظام الذي يتم تجربته وتسليمه للعميل. ولذلك سنشرح طريقة تعريب النظام بصفة تلقائية بمجرد استدعاء البرنامج الرئيسي للتنفيذ عن طريق إضافة أوامر جديدة في بداية البرنامج.

تشغيل النظام بالدمجية العربية

طورت شركة سعودي سوفت برنامج Application Programming Interface وتختصر هكذا API بغرض تسهيل التعامل مع مساعد العربي / ٢ من خلال البرامج التطبيقية التي يعدها المبرمجون. وهذا البرنامج موجود ضمن ملفات قرص المساندة (الذي يأتي ضمن حزمة مساعد العربي). ويتم نقله إلى القرص الثابت أثناء تركيب مساعد العربي / ٢ إذا اتبعت تعليمات التشغيل التي تنصح بها الشركة المنتجة تحت دليل اسمه MA20\API

ويتم التعامل مع برنامج API الخاص بقاعدة البيانات بتحميل برنامج DB_API.BIN (وهذا برنامج مكتوب بلغة التجميع) في الذاكرة ثم استدعائه لأداء وظائف التعريب التي تحتاج إليها من داخل البرنامج ومعروف أن تحميل أحد البرامج المكتوبة بلغة التجميع في الذاكرة يتم بواسطة أمر

LOAD < binary file name >

أما استدعائه للتنفيذ فيتم بأمر

CALL< binary file name>

وفي الحالتين يجب أن يكون البرنامج موجوداً على نفس الدليل أو إضافة اسم الدليل ومشغل القرص قبل اسم البرنامج.

ويتم استدعاء برنامج DB_API.BIN الذي يتحكم في تعريب النظام بالصيغة التالية

CALL DB_API WITH "<function>"

حيث: <function> هي الوظيفة التي تتحكم في برنامج DB_API

ملاحظة: سنوضح أهم الوظائف واختياراتها بعد قليل إلا أننا ننصحك بمراجعة كتيب الشركة المنتجة لمساعد العربي / ٢ لمزيد من التفاصيل عن الوظائف الأخرى.

فمثلاً لكي تستدعي هذا البرنامج ليقلب اتجاه الشاشة من اليمين إلى اليسار وبالتالي تغيير اللغة إلى العربية ومنع قلبها إلى الاتجاه المعاكس بواسطة المفاتيح يتم

استدعاء البرنامج هكذا:

CALL DB_API WITH "SCREEN ARABIC,LOCK"

وننصح بتجميع الأوامر التي تتحكم في تعريب المدخلات والمخرجات داخل إجراء معين ووضعها ضمن ملف الإجراءات تلافياً للتكرار من ناحية وليكون جاهزاً عندك للاستخدام في تطبيقات أخرى.

وقد وضعنا الأوامر التي تتحكم في تعريب المدخلات والمخرجات داخل الإجراء arb_mod والأوامر التي تعيد المدخلات والمخرجات إلى الوضع الأصلي لها داخل الإجراء Lat_mod وكل المطلوب أن تستدعي الإجراء arb_mod في بداية النظام والإجراء Lat_mod قبل إنهاء البرنامج والخروج من النظام.

ويشتمل شكل ١ - ١٠ على محتويات الإجراءين Lat_mod و arb_mod .

```

373 *****
374 * Procedure      : arb_mod
375 * Purpose        : To call DB_API with arabization options
376 *****

377 PROCEDURE arb_mod
378 CALL DB_API WITH 'CODEPAGE ARABIC'
379 CALL DB_API WITH 'NUMER HINDU'
380 CALL DB_API WITH 'SCREEN ARABIC'
381 CALL DB_API WITH 'ACSD ON'
382 CALL DB_API WITH 'DISPLAY VIRTUAL'
383 CALL DB_API WITH 'PRINT_ORIEN ARABIC'
384 CALL DB_API WITH 'PRIM_LANG NO'
385 * ---- Notes:
386 * You may set MA/2 environment by loading a pre-defined macro
387 * with the following syntax:
388 * CALL DB_API WITH 'MACRO <MA macro filename>'
389 * ----
390
391 *****
392 * Procedure      : lat_mod
393 * Purpose        : To call DB_API to reset arabization options
394 *****

395 PROCEDURE lat_mod
396 CALL DB_API WITH 'NUMER ARABIC'

```

شكل ١ - ١٠ الإجراء arb_mod والإجراء Lat_mod

```

397 CALL DB_API WITH 'SCREEN LATIN'
398 CALL DB_API WITH 'PRINT_ORIENT LATIN'
399 CALL DB_API WITH 'PRINT_LANG YES'
400 * ---- Notes:
401 * You may set MA/2 environment by loading a pre-defined macro
402 * with the following syntax:
403 * CALL DB_API WITH 'MACRO <MA macro filename>'
404 * ----
405

```

تابع شكل ١ - ١٠

ونوضح فيما يلي الوظائف التي يشتمل عليها الإجراءان arb_mod و Lat_mod لتعريب النظام واختياراتها.

الوظيفة	اختياراتها
CODE PAGE	ARABIC: تحدد أن نوع الشفرة المستخدمة هو ٧٨٦ عند استخدام الحروف العربية ENGLISH: تحدد أن نوع الشفرة المستخدمة هو ٤٣٧ عند استخدام الحروف الإنجليزية
NUMERALS	HINDU: لإظهار الأرقام بالرسم العربي الموجود على لوحة المفاتيح ARABIC: لإظهار الأرقام بالرسم الإنجليزي الموجود على لوحة المفاتيح
SCREEN	ARABIC: تغيير اتجاه الشاشة ليصبح من اليمين إلى اليسار LATIN: تغيير اتجاه الشاشة ليصبح من اليسار إلى اليمين
ACSD	ON: لإظهار الحروف العربية بالرسم العربي الصحيح عند إدخال بيانات

اختياراتها	الوظيفة
OFF: تلغي اختيار الرسم العربي للحروف العربية الموجودة في لوحة المفاتيح VIRTUAL: لإظهار الحروف العربية عند استخدام الشفرة ٧٨٦ PHYSICAL: لإظهار الحروف الإنجليزية عند استخدام الشفرة ٤٣٧	DISPLAY
ARABIC: لتوجيه الطابعة لتطبع من اليمين إلى اليسار LATIN: لتوجيه الطابعة لتطبع من اليسار إلى اليمين	PRINT_ORIEN
ON: لا تسمح بإعادة لوحة المفاتيح للكتابة باللغة الأصلية بعد ضغط مفتاح الإدخال أو أحد مفاتيح الوظائف الأخرى عند التغيير إلى اللغة الأخرى OFF: تسمح بإعادة لوحة المفاتيح إلى اللغة الأصلية المستخدمة في الكتابة بعد ضغط مفتاح الإدخال أو أحد مفاتيح الوظائف عند التغيير إلى اللغة الأخرى	PRIM_LANG

يشتمل شكل ٢ - ١٠ على برنامج DBAPPARB.PRГ الذي يستدعي القائمة الرئيسية للنظام بعد إعادة كتابة البيانات النصية والرسائل باللغة العربية باستخدام مساعد العربي / ٢. ويفحص هذا البرنامج نجد أنه يزيد عن برنامج DBAPP.PRГ الذي يستدعي قائمة النظام الإنجليزية بالأوامر الآتية.

LOAD DB_API_ (سطر ١١) لتحميل DB_API.BIN في الذاكرة في بداية العمل (ويجب أن تكون نسخت هذا الملف إلى الدليل الحالي).

DO arb_mod_ (سطر ١٢) لاستدعاء الإجراء arb_mod_ (راجع شكل ١ - ١٠).

DO Lat_mod_ (سطر ٥٧، ٦١) لاستدعاء الإجراء Lat_mod_ قبل الخروج من النظام أو عند انتهاء العمل.

```

* -----*
* Program.....: DBAPPARB.PRG*
* Programmer...: Magdi M. Abu Al-Ata*
* Date.....: Dec. 25,1990*
* Purpose.....: To display main menu and do selected option*
*               : in arabic mode*
* Copyright...: Magdi Mohammad Abu Al-Ata*
* -----*

SET PROCEDURE TO dbproc
PRIVATE scrtitle,scrfootr    && Don't use their values in other modules
LOAD DB_API                 && Load arabization utility
DO arb_mod                   && Call tha arabization procedure
DO init
scrtitle = "القائمة الرئيسية"
scrfootr = "====> اختر رقما أو الحرف المفضل"
ok = .T.
DO WHILE ok
  choice = " "
  DO bordarb WITH scrtitle,scrfootr    && Do screen border
  * Draw menu
  @10, 5 SAY "(1) الاعتمادات المستندية"
  DO intense WITH 10,10,"!"
  @11, 5 SAY "(2) المعـ"
  DO intense WITH 11,12,"م"
  @12, 5 SAY "(3) المشتريات"
  DO intense WITH 12,13,"م"
  @13, 5 SAY "(4) المبيعات"
  DO intense WITH 13,15,"ع"
  @14, 5 SAY "(5) حسابات العملاء"
  DO intense WITH 14,10,"ع"
  @15, 5 SAY "(6) ميزان المراجعة"
  DO intense WITH 15,10,"م"
  @16, 5 SAY "(خ) خروج من البرنامج"
  DO intense WITH 16,10,"ع"
  @17, 5 SAY "(ن) نهاية العمل"
  DO intense WITH 17,10,"ن"
  @22,43 GET choice
  READ
  * Do selected option
  DO CASE
    CASE choice = "1".OR. choice = "!"

```

```

DO ordmenu                                && Purchase orders options
CASE choice = "2" .OR. choice = "م"
DO invmenu                                && Inventory options
CASE choice = "3" .OR. choice = "م"
DO purmenu                                && Recording received items
CASE choice = "4" .OR. choice = "ع"
DO salmenu                                && Process sales invoices.
CASE choice = "5" .OR. choice = "ع"
DO cusmenu                                && Process customer data
CASE choice = "6" .OR. choice = "م"
DO mntmenu                                && Maintenance options
CASE choice = "ع"
* Restore system environments before exit
DO reset
* Reset arabization options
DO lat_mod
RETURN                                    && Go to Dot-Prompt
CASE choice = "ن"
CLEAR
DO lat_mod                                && Reset arabization options
QUIT                                     && Quit to DOS
OTHERWISE
LOOP
ENDCASE
ENDDO
*****
* End of file DBAPPARB.PRG *
*****

```

تابع شكل ٢ - ١٠ برنامج DBAPPARB.PRG

وقد استبدلنا الإجراء border الذي يرسم شاشة البداية؛ بالإجراء bordarb لرسم الشاشة باللغة العربية ويشتمل شكل ٣ - ١٠ على محتويات هذا الإجراء وفيه تلاحظ أننا استبدلنا النصوص الموجودة بالبرنامج بنظيرتها العربية باستخدام مساعد العربي / ٢.

ويشتمل شكل ٤ - ١٠ على القائمة التي تنتج من تنفيذ برنامج

DBAPPARB

```

357
358 *****
359 * Procedure : bordarb
360 * Purpose : Draws a screen frame
361 *****
362 PROCEDURE bordarb
363 PARAMETERS scrttitle,scrfootr
364 PUBLIC systitle,title2,mdate,mtime
365 systitle = "شركة الحاسب العربى"
366 title2 = "ص.م. 39387"
367 mdate = "DTOC( DATE()) الحاربيخ:"
368 mtime = "TIME() الوقت:"
369 CLEAR
370 DO intense WITH 0, 1, mdate
371 DO intense WITH 0,78 - LEN(mtime), mtime
372 IF mcolor
373 | SET COLOR TO R+/N
374 ELSE
375 | SET COLOR TO W+/N
376 ENDIF
377 @ 1, 1 TO 4,78 DOUBLE
378 @ 7, 1 TO 23,78 DOUBLE
379 @ 2,80/2-(LEN(systitle)/2) SAY systitle
380 @ 3,80/2-(LEN(title2)/2) SAY title2
381 @ 8,80/2-(LEN(scrttitle)/2) SAY scrttitle
382 @ 9,2 TO 09,77
383 @21,2 TO 21,77
384 @22,5 SAY scrfootr
385 SET COLOR TO W+/N
386 RETURN ** End of procedure
387
388 *****

```

شكل ٣ - ١٠ الإجراء bordarb

وهكذا تستطيع تعريب النظام كله بتغيير النصوص والرسائل الموجودة داخل البرامج فقط إلى نظيرتها باللغة العربية.

اقتراحات لتيسير كتابة البرامج

أثناء كتابة البرامج نحتاج لكتابة بيانات عربية وبيانات لاتينية في نفس السطر

الشاريخ: ٩١/٠٦/٠٥	الوقت: ١٠:٢٤:٠٥		
شركة الحاسب العربي ص.٥: ٣٩٣٨٧			
القائمة الرئيسية			
<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%;"></td> <td style="width: 50%; text-align: right;"> (١) الاعتمادات المستندية (٢) المذونان (٣) المشتريات (٤) المبيعات (٥) حسابات العملاء (٦) ميزان المصروفات (٧) خروج من البرنامج (٨) نهاية العمل </td> </tr> </table>			(١) الاعتمادات المستندية (٢) المذونان (٣) المشتريات (٤) المبيعات (٥) حسابات العملاء (٦) ميزان المصروفات (٧) خروج من البرنامج (٨) نهاية العمل
	(١) الاعتمادات المستندية (٢) المذونان (٣) المشتريات (٤) المبيعات (٥) حسابات العملاء (٦) ميزان المصروفات (٧) خروج من البرنامج (٨) نهاية العمل		
اختر رقما أو الحرف المفضل <=====			

شكل ٤ - ١٠ قائمة النظام المعربة

مما يسبب صعوبة كبيرة في التعامل مع الشاشة واتجاهاتها. لذلك ننصحك بتغيير بعض اختيارات مركز الإشراف لتناسب هذا الوضع. ونقترح تخزين هذه الاختيارات تحت «ماكرو» يمكنك استدعائه في المرات القادمة عند الحاجة وذلك باتباع الآتي:

- اضغط مفتاحي Alt + Num Lock لتجعل اتجاه الشاشة من اليمين إلى اليسار لكي تظهر البيانات باللغة العربية يمكنك متابعة الشرح.
- اضغط مفتاحي Alt + Scroll Lock لاستدعاء مركز الإشراف ستظهر البيانات باللغة العربية وسيظهر سطر القوائم وبه أربعة اختيارات تحت كل منها قائمة منسدلة والاختيارات الأربعة هي:

الوضع خيارات طباعة خدمات

- استخدم مفاتيح الأسهم → ← للتنقل إلى الاختيار خيارات.

- استخدم مفتاح السهم ↓ لاختيار تمكين اللغة ألياً ثم غير نعم إلى لا (التبديل يتم بضغطة مسطرة المسافات).
- اضغط مفتاح ESC للعودة إلى القائمة.
- اختر إنهاء انسياب اللغة الثانوية واختر نعم أمام كل من عند الأرقام وعند المحايد.
- اختر Esc للعودة إلى القائمة.
- اختر خدمات ثم ماكرو ثم خزن ماكرو ثم اكتب اسم الدليل واضغط مفتاح الإدخال ثم اختر اسماً للماكرو وتذكر هذا الاسم ليتمكنك استدعائه به في المرات القادمة.
- اضغط مفتاح ESC للعودة للقائمة وللخروج من مركز الإشراف.
- عند تشغيل مساعد العربي / ٢ في المرة القادمة افتح مركز الإشراف واختر خدمات ثم ماكرو ثم حمل ماكرو ثم اكتب اسم الدليل واختر الماكرو المطلوب.

تعريب النظام باستخدام نافذة

برنامج نافذة واحد من أشهر البرامج التي أخذت على عاتقها تعريب مدخلات ومخرجات الحاسب الآلي. وهو من إنتاج شركة 01 المنامة - البحرين. وتعتمد فكرة نافذة على تحميل البرنامج في ذاكرة الحاسب بحيث يبقى بالذاكرة حتى يتم إغلاق الحاسب. وذلك لتسهيل التعامل مع معظم البرامج التطبيقية.

تشغيل نافذة

لتشغيل نافذة ضع قرص البرنامج في مشغل الوحدة A أو B ثم اكتب
NAF > A واضغط مفتاح الإدخال. فإذا أردت تشغيل نافذة وتحميل مشغل الطباعة المتصلة بالحاسب اكتب A:\NAFITHA ثم اضغط مفتاح الإدخال.

سيتولى البرنامج تحميل حروف الطباعة في ذاكرة الحاسب بالإضافة إلى تحميلها في ذاكرة الطباعة. أما إذا كانت نافذة محملة على القرص الثابت فيجب أولاً الانتقال

إلى الدليل الذي توجد تحته نافذة بأمر CD ثم كتابة NAF أو NAFITHA .
فإذا حملت نافذة بدون مشغل الطابعة وأردت بعد ذلك - أثناء تحميل نافذة -
طباعة بيانات عربية أدر مفتاح تشغيل الطابعة ثم اكتب LOADNAF .
ملاحظة: إذا كان ملف CONFIG.SYS متضمناً ملف تشغيل الشاشة القياسي
ANSI.SYS اكتب NAF/A أو NAFITHA/A تحت عث «الدوس» ثم اضغط مفتاح
الإدخال.

اختيار اتجاه الكتابة على الشاشة واللغة

هناك حالتان لاتجاه الكتابة على الشاشة. الأولى: اتجاه الكتابة من اليسار إلى
اليمين وهو الوضع الأصلي للكتابة بعد تحميل نافذة مباشرة. الثانية: اتجاه الكتابة
من اليمين إلى اليسار وهو الاتجاه الذي تتيحه نافذة. كذلك يمكن الكتابة بالحروف
اللاتينية كما يمكن الكتابة بالحروف العربية. والكتابة بالحروف اللاتينية هي الوضع
الأصلي للوحة المفاتيح والوضع الذي تكون عليه لوحة المفاتيح بعد تحميل نافذة
مباشرة.

وليس هناك أي ارتباط بين اختيار اتجاه الكتابة على الشاشة واللغة المستخدمة
للكتابة فيمكن اختيارهما في أي تركيبة ترغبها حسب حاجتك وحسب متطلبات
البرنامج الذي تستخدمه. فمثلاً في حالة استخدام برنامج إنجليزي يجب اختيار
اتجاه الكتابة على الشاشة من اليسار إلى اليمين، أما في حالة استخدام برنامج عربي
فيفضل الكتابة العربية واختيار اتجاه الكتابة على الشاشة من اليمين إلى اليسار
ويوضح الجدول التالي مجموعة من الحالات التي يتم فيها اختيار اتجاه الكتابة على
الشاشة واللغة مع نافذة.

الاختصار	المفاتيح المستخدمة
اختيار الاتجاه العربي	(التحكم + عالي اليمين) Ctrl + Right Shift
اختيار الاتجاه اللاتيني	(التحكم + عالي اليسار) Ctrl + Left Shift
اختيار الحروف العربية	(البديل + عالي اليمين) Alt + Right Shift
اختيار الحروف اللاتينية	(البديل + عالي اليسار) Alt + Left Shift

تشغيل النظام بالطريقة العربية

يمكن تشغيل النظام بالطريقة التي شرحناها وهي تغيير اتجاه الشاشة واللغة بواسطة لوحة المفاتيح إلا أننا لا ننصح بهذا الأسلوب خصوصاً بالنسبة للنظام الذي يتم تجربته وتسليمه للعميل. ولذلك سنشرح كيفية تشغيل النظام بالطريقة العربية عن طريق إضافة أوامر جديدة في بداية البرنامج.

سنعمد لإضافة إجراءات جديدين الأول لتمكين تعريب الشاشة واللغة والثاني عكس ذلك ويمكن استدعاء أحدهما عند الحاجة إليه كما فعلنا عند شرح تعريب النظام بواسطة مساعد العربي / ٢.

يشتمل شكل ٥ - ١٠ على الإجراءات arb_con و Lat_con

واليك شرح الأوامر الواردة بكل منهما.

- أولاً: الإجراء arb_con وينفذ في بداية النظام لتعريب الشاشة ولوحة المفاتيح.
- سطر ٤٢١ لتغيير اتجاه الشاشة ليصبح من اليمين إلى اليسار.
- سطر ٤٢٢ لتغيير لغة الكتابة لتصبح عربية.
- سطر ٤٢٤ وضعنا أمامه علامة * لتعطيله مؤقتاً وتركناه لتستفيد منه إذا رغبت في ذلك ومهمته تعطيل تغيير اتجاه الشاشة بواسطة المفاتيح. ونلجأ إلى ذلك في النظم التي تنتهي تماماً وتسلم للعميل ونخشى من سوء استخدام المستفيد للمفاتيح مما يغير في اتجاه الشاشة.

```
436 *****
437 * Procedure : arb_con
438 * Purpose   : To arabize screen and keyboard
439 *****

440 PROCEDURE arb_con
441 ?CHR(27)+"fa"      && Set arabic screen orientation
442 ?CHR(27)+"fe"      && Set keyboard language to arabic
443 *
444 * ?CHR(27)+"fo0"    && Disable screen orientation switching by use
445 * ?CHR(27)+"fk0"    && Disable keyboard language switching by use
446 RETURN
447
448 *****
449 * Procedure : lat_con
450 * Purpose   : To reset arabizing screen and keyboard
451 *****

452 PROCEDURE lat_con
453 ?CHR(27)+"fl"      && Set english screen orientation
454 ?CHR(27)+"fk"      && Set keyboard language to latin
455 *
456 * ?CHR(27)+"fo1"    && Enable screen orientation switching by use
457 * ?CHR(27)+"fk1"    && Enable keyboard language switching by use
458 RETURN
459
```

شكل ٥ - ١٠ الإجراء arbcon والإجراء Latcon

- سطر ٤٤٥ أيضًا ملغي هنا ومهمته تعطيل تغيير لغة الكتابة.
- ثانيًا: الإجراء lat_con وينفذ عندما نرغب في إنهاء البرنامج أو إنهاء العمل ومهمته إرجاع الشاشة ولوحة المفاتيح للوضع اللاتيني الأصلي.
- سطر ٤٥٣ لتغيير لغة الكتابة لتصبح إنجليزي.
- سطر ٤٥٦ لتمكين تغيير اتجاه الشاشة بواسطة المفاتيح . وهو ملغي هنا.
- سطر ٤٥٧ لتمكين تغيير لغة الكتابة بواسطة المفاتيح وهو أيضًا ملغي هنا . وقد أوردناه لتوضيح مهمته عند الحاجة إليه.

لتنفيذ النظام باستخدام برنامج نافذة للتعريب استبدل اسم الإجراءين lat_mod و arb_mod بالإجراءين arb_con و lat_con في برنامج DBAPPARB.PRG الذي شرحناه في شكل ٢ - ١٠ السابق.

ثم أعد كتابة النصوص والرسائل الموجودة في البرامج باستخدام محرر السطور الذي يروق لك بعد تحميل نافذة ستحصل على نفس القائمة العربية التي حصلنا عليها في شكل ٤ - ١٠ السابق.

ملاحظة: لاحظ أن نافذة تستخدم شفرة مختلفة عن شفرة مساعد العربي / ٢. ولذلك تعريب مساعد العربي / ٢ لا يصلح مع نافذة والعكس صحيح إلا أن مساعد العربي / ٢ توجد به إمكانية التعامل مع بيانات مكتوبة بنافذة وذلك باختيار نظام المحاكاة المناسب.

الفصل السادس عشر

تعقب وتصحيح أخطاء البرامج

Debugging

من البديهيات المهمة أن المبرمج مهما كان خليقاً ومهتماً فلا بد أن يقع في أخطاء ولذلك فإن عملية اكتشاف الأخطاء وتصحيحها قبل تجربة النظام أو تسليمه للعميل لا تقل في أهميتها عن كتابة البرنامج أو النظام نفسه.

ولا يوجد في قاعدة البيانات dBASE III PLUS مكتشف للأخطاء (Debugger) مثل dBASE IV أو Clipper ولذلك فسنحاول في هذا الفصل شرح الوسائل التي تساعد في تعقب واكتشاف أخطاء البرامج عن طريق بعض الأوامر أو تغيير بعض الدوال.

تسمى أخطاء البرنامج بلغة الحاسب "Bugs" وتسمى عملية تعقب الأخطاء وتنقيحها "Debugging" ويمكن تقسيم أخطاء البرنامج عمومًا إلى نوعين من الأخطاء.

أولاً: أخطاء لغوية أو نحوية. وهي التي تسبب توقف البرنامج ولا يمكن تنفيذ البرنامج بدون تصحيح هذه الأخطاء.
ثانياً: أخطاء منطقية. وهذه الأخطاء لا تظهر إلا في مرحلة التجارب لأنها لا تسبب توقف البرنامج إلا أنها تعطي نتائج غير سليمة وفيما يلي سناقش هذين النوعين من الأخطاء بالتفصيل.

أولاً: الأخطاء النحوية:

هذا النوع من الأخطاء هو الذي يكتشفه وينبه إليه المفسر الموجود في قاعدة البيانات ويسمى Interpreter وهو خاص برصد أخطاء قواعد اللغة وتركيبها ويرجع إلى أسباب عديدة من أمثلتها الأخطاء الإملائية في كتابة أحد الأوامر كأن تكتب LST بدلاً من LIST أو APEND بدلاً من APPEND ، أو الخطأ في تركيب أحد الأوامر (Syntax error). وكذلك نسيان تعريف حقل أو متغير معين، أو استخدام إحدى العلامات الممنوعة.

وتتولى قاعدة البيانات اكتشاف هذه الأخطاء والتنبيه عنها أثناء تنفيذ البرنامج وتطلب منك تصحيح الخطأ ولا يمكن تنفيذ أي برنامج ما لم يكن خالياً من هذه الأخطاء. ونوضح فيما يلي مثلاً لهذا النوع من الأخطاء.

إذا كتبت أمر APPEND BLANK في برنامج إضافة عميل جديد CUSADD.PRG على سبيل الخطأ هكذا APEND BLANK فسوف توقف قاعدة البيانات تنفيذ البرنامج وستظهر لك الرسالة التالية:

*** Unrecognized command verb.

?

APEND BLANK

Called from - C:\cusadd.prg

Called from - C:\cusmenu.prg

Called from - C:\DBAPP.prg

Cancel, Ignore, or Suspend? (C, I, or S)

إذا اخترت C ردًا على الرسالة Cancel فسيتم إلغاء البرنامج والعودة إلى نقطة المحدث. أما إذا اخترت I بمعنى Ignore فستهمل قاعدة البيانات الأمر الذي حدث به خطأ ثم تستمر في تنفيذ باقي البرنامج. إذا اخترت S (Suspend) فستعود إلى نقطة المحدث إلا أن البرنامج سيظل موجودًا بالذاكرة ولذلك يمكنك فحص المتغيرات الخطأ أو تصحيح الخطأ الذي ظهر لأن قاعدة البيانات لا تظهر فقط الأمر الخطأ وإنما تظهر لك أيضًا اسم البرنامج الذي وقع به الخطأ داخل النظام والبرامج الأخرى التي تستدعيه (إذا كان الخطأ موجودًا داخل ملف إجراءات فلن تحرك قاعدة البيانات عن اسم الإجراء الذي يوجد به الأمر ولكنها ستخبرك عن اسم ملف الإجراءات الذي يشتمل على كل الإجراءات). فإذا أردت العودة لتنفيذ البرنامج من النقطة التي توقف عندها أدخل أمر RESUME من نقطة المحدث.

ثانيًا: الأخطاء المنطقية:

في هذا النوع من الأخطاء يصير البرنامج خاليًا من الأخطاء وينفذ طبيعيًا بدون رسائل خطأ إلا أن النتائج التي تحصل عليها تختلف عن تلك المتوقعة من البرنامج. ومن أمثلة ذلك أن يتم طرح بدل الانتقال من الراتب الأساسي بدلًا من جمعه إليه في برنامج حساب المرتبات. وقد تختلف النتائج عن تلك المتوقعة نتيجة لتحليل خاطئ للنظام منذ البداية.

ومثل هذا النوع من الأخطاء قد يستغرق اكتشافها الكثير من الوقت والجهد. ولتقليل هذه الأخطاء يجب أن يتم تصميم النظام بعناية شديدة ومراجعة خرائط النظام ومواصفات الملفات جيدًا قبل بدء عملية البرمجة.

وتستدعي عملية تعقب واكتشاف الأخطاء المنطقية حساسية معينة والنظر إلى البرنامج أو النظام كله بدقة وعمق لأنها لا تكون مباشرة.

ونوضح فيما يلي الوسائل التي تساعد في تعقب اكتشاف أخطاء البرنامج.

تغيير حالة بعض الدوال

أمر SET ECHO من الدوال التي تستخدم في حالتين ON و OFF والوضع التلقائي لها هو OFF . فإذا حصلت على مشكلة تحتاج لتوضيح أكثر استخدم أمر SET ECHO ON لتظهر لك أوامر البرنامج على الشاشة أثناء التنفيذ . ولأن الأوامر تظهر على الشاشة بسرعة قد لا تستطيع ملاحظتها . فإذا أردت أن ينفذ البرنامج أمراً وأمرًا وأن يتوقف التنفيذ بعد كل أمر مؤقتاً حتى تتمكن من قراءة الأوامر أضف إلى البرنامج أمر:

SET STEP ON

ستظهر مع كل أمر الرسالة التالية :

PRESS SPACE to step, S to suspend. or Esc to Cancel...

وإذا أردت توجيه المخرجات إلى الطابعة لتسهيل مراجعتها استخدم أمر:

SET DEBUG ON

ونود الإشارة إلى أن البرنامج بعد انتهاء مرحلة التجارب يجب أن يخلو من هذه الأوامر الثلاثة :

SET ECHO ON

SET STEP ON

SET DEBUG ON

إضافة أوامر لتعقب واكتشاف الأخطاء

تشتمل قاعدة البيانات على بعض الأوامر التي تستخدم لغرض تعقب واكتشاف الأخطاء وننصح أيضاً ألا يشتمل عليها البرنامج بعد انتهاء مرحلة التجارب . هذه الأوامر هي :

١ - أمر ON ويأخذ إحدى الصيغ الآتية :

ON ERROR [<Command>]

ON ESCAPE [<Command>]

ON KEY [<Command>]

ويتسبب هذا الأمر في تنفيذ أحد الأوامر (<Command>) عند حدوث واحدة من الحالات الثلاث التالية :

١ - عند اكتشاف قاعدة البيانات خطأ ما (ON ERROR) .

٢ - عند ضغط مفتاح Esc (ON ESCAPE) .

٣ - عند ضغط أي مفتاح (ON KEY) .

ويمكن استخدام الوظيفة () ERROR داخل البرنامج لتحديد نوع معين من الأخطاء واتخاذ إجراء بشأنه. انظر المثال التالي :

الأمر التالي يفترض وجود برنامج على الدليل الحالي اسمه FINDERR.PRG

ON ERROR DO FINDERR

وبالتالي فعند اكتشاف قاعدة البيانات لأي خطأ فستنتقل إلى البرنامج التالي :

```
* Program: FINDERR.PRG
IF ERROR() = 6                && Too many files are open
  WAIT "The clause FILES = 20 must be in DOS file CONFIG.SYS"
  ONKEY QUIT
ENDIF
RETURN
```

والوظيفة () ERROR تشتمل على رقم رسالة الخطأ ولما كان الرقم 6 يعادل الرسالة التالية :

"Too Many files are open"

فمعنى ذلك أن الوظيفة () ERROR ستشتمل على الرقم 6 إذا حاولت أن تفتح أكثر من عدد الملفات المسموح بها.

٢ - أمر SUSPEND

يسبب الخروج من البرنامج إلى نقطة المحث مع الاحتفاظ بالبرنامج داخل ذاكرة الحاسب ولاستئناف تنفيذه مرة ثانية أدخل أمر RESUME .

٣ - أمر DISPLAY STATUS أو DISPLAY MEMORY

يظهر أمر DISPLAY MEMORY محتويات الذاكرة في لحظة معينة داخل البرنامج وبالتالي يمكن أن يساعدك فحص محتويات الذاكرة على التعرف على المشكلة كما يظهر أمر DISPLAY STATUS المعلومات المتوفرة عن الملفات المفتوحة وملفات الفهرسة ومفاتيح الفهرسة والدوال المستخدمة . ويمكنك توجيه نتائج هذين الأمرين إلى ملف نصي (TXT) بدلاً من إظهارها على الشاشة وإرباك البرنامج كله بإصدار الأمرين التاليين قبل أمر DISPLAY MEMORY في المكان الذي تريد عنده الاطلاع على محتويات الذاكرة

SET ALTERNATE TO <file name>

SET ALTERNATE ON

ويمكنك تكرار هذه الأوامر في أكثر من موضع داخل البرنامج والاطلاع على محتويات الذاكرة أو معرفة حالة النظام بطباعة الملف النصي الذي أنشأته .

الملاحق

الملحق الأول: النطرق الأمريكية (ASCII)

الملحق الثاني: الحزم البرامقية المتوافقة مع

«دي بي إس»

الملحق الثالث: ملف الإجراءات DBPROC.PROG

الملحق الأول

الشفرة الأمريكية القياسية لتبادل المعلومات (ASCII)

يوضح الجدول التالي قيم الرموز والأرقام والأحرف والعلامات الخاصة المقابلة للشفرة الأمريكية (ASCII Code) مرتبة تصاعديا حسب القيمة العشرية المقابلة لكل حرف أو رقم أو رمز يوضح الجدول كذلك معنى الأكواد من صفر إلى ٣١

وتستخدم وظيفة () ASC في قاعدة البيانات للحصول على الشفرة المقابلة للحروف. فإذا أردت معرفة الشفرة (ASCII Code) المقابلة للحرف M أدخل الأمر التالي :
? ASC ("M")

ستحصل على الرقم 77

بينما تستخدم وظيفة () CHR لظهار الحرف المقابل للشفرة (ASCII Code) التي تعطى له. فمثلا لمعرفة الحرف المقابل للعدد 77 يجب أن تدخل الأمر الآتي

? CHR (77)

ستحصل على حرف M

ASCII Value	Character	Code	Symbol
الشفرة	الحرف	الكود	الرمز
000	(null)	(null)	NUL
001	~	Ctrl-A	SOH
002	●	Ctrl-B	STX
003	♥	Ctrl-C	ETX
004	♦	Ctrl-D	EOT
005	♣	Ctrl-E	ENO
006	♠	Ctrl-F	ACK
007	(beep)	Ctrl-G	(bell) BEL
008	■	Ctrl-H	(backspace) BS
009	(tab)	Ctrl-I	(tab) horizontal HT
010	(line feed)	Ctrl-J	(linefeed) LF
011	(home)	Ctrl-K	(vertical tabs) VT
012	(form feed)	Ctrl-L	(formfeed) FF
013	(carriage return	Ctrl-M	(carriage return) CR
014	♪	Ctrl-N	SO
015	⚙	Ctrl-O	SI
016	▶	Ctrl-P	DLE
017	◀	Ctrl-Q	DC1
018	⬆	Ctrl-R	DC2
019	!!	Ctrl-S	DC3
020	π	Ctrl-T	DC4
021	Ⓜ	Ctrl-U	NAK
022	—	Ctrl-V	SYN
023	⬆	Ctrl-W	ETB
024	⬆	Ctrl-X	CAN
025	⬆	Ctrl-Y	EM
026	→	Ctrl-Z	SUB
027	←	Escape	
028	(cursor right)	FS	
029	(cursor left)	GS	
030	(cursor up)	RS	
031	(cursor down)	US	

ASCII Value الشفرة	Character الحرف	ASCII Value الشفرة	Character الحرف
032	(space)	069	E
033	!	070	F
034	"	071	G
035	#	072	H
036	\$	073	I
037	%	074	J
038	&	075	K
039	'	076	L
040	(077	M
041)	078	N
042	*	079	O
043	+	080	P
044	,	081	Q
045	.	082	R
046	.	083	S
047	/	084	T
048	0	085	U
049	1	086	V
050	2	087	W
051	3	088	X
052	4	089	Y
053	5	090	Z
054	6	091	[
055	7	092	\
056	8	093]
057	9	094	^
058	:	095	_
059	:	096	`
060	^	097	a
061	=	098	b
062	>	099	c
063	?	100	d
064	@	101	e
065	A	102	f
066	B	103	g
067	C	104	h
068	D	105	i

الملحق الأول: الشفرة الأمريكية (ASCII)

ASCII Value الشفرة	Character الحرف	ASCII Value الشفرة	Character الحرف
106	j	143	À
107	k	144	É
108	l	145	æ
109	m	146	Æ
110	n	147	ô
111	o	148	ó
112	p	149	õ
113	q	150	û
114	r	151	ü
115	s	152	v
116	t	153	Ö
117	u	154	U
118	v	155	é
119	w	156	£
120	x	157	¥
121	y	158	₪
122	z	159	₹
123	!	160	¢
124	"	161	£
125	£	162	ó
126	~	163	ü
127	☐	164	ñ
128	Ç	165	Ñ
129	u	166	é
130	é	167	ó
131	à	168	ó
132	à	169]
133	à	170]
134	à	171	½
135	ç	172	¼
136	ê	173	ı
137	e	174	ı
138	ğ	175	ı
139	ı	176	ı
140	ı	177	☒
141	ı	178	☒
142	A	179	ı

ASCII Value الشفرة	Character الحرف	ASCII Value الشفرة	Character الحرف
180	،	218	
181		219	■
182	┐	220	▀
183	┌	221	▄
184		222	▥
185	└┐	223	▦
186		224	α
187	⌒	225	π
188	⌒	226	π
189	┐	227	π
190	┐	228	π
191	┐	229	π
192	┐	230	μ
193	┐	231	~
194	┐	232	¢
195	┐	233	⊕
196	┐	234	⊖
197	┐	235	δ
198	┐	236	α
199	┐	237	⌒
200	┐	238	(
201	┐	239)
202	┐	240	≡
203	┐	241	·
204	┐	242	≥
205	┐	243	≤
206	┐	244	(
207	┐	245)
208	┐	246	÷
209	┐	247	≈
210	┐	248	
211	┐	249	•
212	┐	250	•
213	┐	251	√
214	┐	252	n
215	┐	253	:
216	┐	254	■
217	┐	255	(blank 'FF')

الملحق الثاني

الحزم البرمجية المتوافقة مع dBASE

Add_on Packages

نورد فيما يلي أسماء وعناوين أهم الحزم البرمجية المتوافقة مع dBASE III PLUS والحزم الأخرى التي تسهل تطوير نظم إدارة قواعد البيانات.

■ **FoxBase**: هي من إنتاج شركة «فوكس سوفتوير» الأمريكية وطورت إلى FoxBase + وأخيراً FoxPr وهي أكثر قواعد البيانات توافقاً مع «دي بيس ثري بلاس».

عنوان الشركة المنتجة:

Fox Software, Inc.

134W. South Boundry Tel: (419) 874-0162

U.S.A.

Perrysburg, Ohio 43551 Fax: (419) 874-8678

■ **dBXL**: ومن من إنتاج شركة «ورد تيتش سيستمز» وهي أيضاً من أكثر قواعد البيانات توافقاً مع «دي بيس ثري بلاس» وتتميز عنها باستخدام النوافذ والرسوم البيانية.

عنوان الشركة المنتجة:

Word Teach Systems Inc.

P.O.BOX 1747, Orinda, CA 94563

Tel: (415) 254-0900 Fax: (415) 254-0288

■ **Quicksilver**: من إنتاج شركة «كويك سلفر سوفتوير» واشترت حقوقها شركة «ورد تيتش سيستمز» وهي عبارة عن مترجم للغة دي بيس متوافق إلى حد كبير مع «دي بيس ثري بلاس» ومتوافق تماماً مع «دي بي إكس إل» وبإمكانه

استخراج برامج جاهزة للتنفيذ. (.EXE) إلا أنه لا يشتمل على مفسر (Interpreter) مثل «دي بيس ثري بلاس». عنوان الشركة المنتجة:

Word Teach Systems Inc.

P.O.BOX 1747, Orinda, CA 94563

Tel: (415) 254-0900 Fax: (415) 254-0288

■ Clipper: من إنتاج «نانتوكيت» الأمريكية وأخذت شهرة عظيمة نظراً للإمكانيات التي تشتمل عليها والتي تعتبر إضافة إلى قاعدة البيانات dBASE III PLUS مثل استخدام المصفوفات والوظائف الخاصة والقوائم ذات الشريط المضاء (light bar) وتعتبر أفضل مترجم (Compiler) للغة dBASE وتستخرج ملفاً جاهزاً للتنفيذ (.EXE) إلا أنها تشتمل على أوامر كثيرة غير موجودة في «دي بيس» كما أنها لا تتعامل مع كثير من أوامر «دي بيس».

[راجع كتابنا المرجع الأساسي لقاعدة البيانات Clipper].

عنوان الشركة المنتجة:

Nantucket Corporation,

125555 West Jefferson Boulevard, Los Angeles, CA 90066

U.S.A.

Tel: (213) 390-7923 FAX: (213) 397-5469

■ R & R: برنامج Relational Report Writer وتختصر R & R من إنتاج شركة «كوسنترك داتا سيستمز».

ومهمته استخراج تقارير من ملفات قاعدة البيانات بكفاءة وسهولة وسرعة عالية تزيد عن إمكانيات استخراج التقارير بواسطة مصمم التقارير الموجود في قاعدة البيانات dBASE III PLUS. والبرنامج يستطيع استخراج تقرير واحد من

عشرة ملفات مختلفة بعد ربطها معًا. ويمكنه كذلك استخراج ملصقات بريدية.
ويمكن تشغيل البرنامج تحت عت نظام التشغيل DOS أو تحت قاعدة البيانات
dBASE III PLUS

عنوان الشركة المنتجة:

Concentric Data Systems, Inc.

110 Turnpike Road, Westboro, MA 01581

U.S.A.

Tel: (800) 325-9035 FAX: 508-366-2954

■ **Genifer**: برنامج مفيد للمبتدئين وقليل الخبرة بالبرمجة وهو يستخدم لتصميم
تطبيقات وكتابة برامج ويمكن تعديل البرامج التي يكتبها لتوفيقها حسب
رغبتك. وهو يوفر كثيرًا من الوقت اللازم لإعداد القوائم والتقارير وشاشات
الإدخال والتعديل.

عنوان الشركة المنتجة:

Bytel Crop. 1029 Solano Ave.

Berkeley, CA 94706

U.S.A.

Tel: (415) 527-1157

■ **QUICKCODE**: يستخدم لنفس الغرض الذي يستخدم من أجله برنامج
Genifer السابق.

عنوان الشركة المنتجة:

Fox and Geller

504 Market Street Elmoood Park, NJ 07407

U.S.A.

Tel: (201) 794-9993

■ CLEAR + :برنامج مفيد لاستخراج خرائط النظم وخرائط التدفق وطباعة البرامج بطريقة منظمة.
عنوان الشركة المستفيدة:

CLEAR Software Inc.

1440 Beacom Street, Suit 1,

Brookline, MA 02146

U.S.A.

Tel: (617) 232-4720 FAX (617) 232-9788

■ dFLOW :برنامج مفيد في عمل تحليلات للبرامج ومعرفة علاقاتها ببعضها داخل النظام وطباعتها بطريقة منظمة.
عنوان الشركة المنتجة:

Wall Soft Systems 233 Braodway

New York NY 10279

(800) 23d_Flow

الملحق الثالث

ملف الاجراءات DBPROC.PRG

```

1 * ----- *
2 * Program.....: DBPROC.PRG *
3 * Programmer...: Magdi M. Abu Al-Ata *
4 * Date.....: Dec. 25,1990 *
5 * Purpose.....: Main procedure file for the system *
6 * ----- *
7
8 *****
9 * Procedure : init
10 * Purpose : Set working environments
11 *****
12 PROCEDURE init
13 PUBLIC msg,mcolor
14 SET SCOREBOARD OFF                && Turn off dBASE message in row 0
15 SET TALK OFF                      && Turn off talking
16 SET STATUS OFF                   && Turn off status bar
17 SET CONFIRM ON                   && Press Enter to accept entry
18 SET BELL OFF                     && Turn bell sound off
19 SET EXACT ON                     && Turn the bell off
20 SET DATE BRITISH                 && Display date in format: dd/mm/yy
21 SET DELETED ON                  && Ignire records marked for deletion
22 CLOSE ALL                       && Close all files
23 msg = " Press " + chr(17)+chr(196)+chr(217) + "key to try again."
24 IF ISCOLOR()
25 |   mcolor = .T.
26 ENDIF
27 IF mcolor
28 |   SET COLOR TO W+/N,W+/G+
29 ENDIF
30 RETURN                          && End of procedure
31
32 *****
33 * Procedure : reset
34 * Purpose : Reset working environments and exit
35 *****
36 PROCEDURE reset
37 SET SCOREBOARD ON
38 SET |ALK ON
39 SET STATUS ON
40 SET CONFIRM OFF
41 SET BELL ON
42 SET EXACT OFF
43 SET DATE AMERICAN
44 CLEAR ALL                       && Erase memory
45 CLEAR                          && Clear screen
46 RETURN                         && End of procedure
47
48 *****
49 * Procedure : border
50 * Purpose : Drows a screen frame
51 *****

```

```

52 PROCEDURE border
53 PARAMETERS scrtitle,scrfootr
54 PUBLIC systitle,title2,mdate,mtime
55 systitle = "Arabian Computer Center"
56 title2   = "C.R. 39387 "
57 mdate    = CDATE(DATE())+" "+STR(DAY(DATE()),2)+" "+MONTH(DATE()) ;
58 + " "+STR(YEAR(DATE()),4)
59 mtime    = "Time: " + TIME()
60 CLEAR
61 DO intense WITH 0, 1, mdate
62 DO intense WITH 0,78 - LEN(mtime), mtime
63 IF mcolor
64 | SET COLOR TO G+/N
65 ELSE
66 | SET COLOR TO W+/N
67 ENDIF
68 @ 1, 1 TO 4,78 DOUBLE
69 @ 7, 1 TO 23,78 DOUBLE
70 @ 2,80/2-(LEN(systitle)/2) SAY systitle
71 @ 3,80/2-(LEN(title2)/2) SAY title2
72 @ 8,80/2-(LEN(scrtitle)/2) SAY scrtitle
73 @ 9,2 TO 09,77
74 @21,2 TO 21,77
75 @22,5 SAY scrfootr
76 SET COLOR TO W+/N
77 RETURN && End of procedure
78 *****
79 * Procedure : intense
80 * Purpose : To display texed in red color
81 *****

82 PROCEDURE intense
83 PARAMETERS mrow,mcol,mstr
84 IF mcolor
85 | SET COLOR TO GR+/N
86 ELSE
87 | SET COLOR TO W+/N
88 ENDIF
89 @ mrow,mcol SAY mstr
90 IF mcolor
91 | SET COLOR TO W+/N
92 ELSE
93 | SET COLOR TO W+/N
94 ENDIF
95 RETURN
96 *****
97 * Procedure : error
98 * Purpose : To display an error message and pause execution
99 *****

101 PROCEDURE error
102 PARAMETER ermsg
103 IF mcolor
104 | SET COLOR TO R+/N
105 ELSE
106 | SET COLOR TO W+/N
107 ENDIF

```

الملحق الثالث : ملف الاجراءات DBPROC.PRG

```

108 @ 24,1 SAY ermsg + msg
109 READ
110 @ 24,1
111 SET COLOR TO W+/N
112 RETURN
113
114 *****
115 * Procedure : ord_fmt
116 * Purpose : Drows order screen format
117 *****

118 PROCEDURE ord_fmt
119 PUBLIC m_ord_no, m_o_val, m_o_knd, m_o_dat, m_o_des, m_sh_nm, ;
119 m_co_rm, m_co_ad, m_b_rm, m_b_ad, m_bl_no, m_doc_no, m_comm, ;
119 m_post, m_ins, m_custom, m_lbr, m_total
120 * Give those variables their starting values
121 STORE SPACE(5) TO m_ord_no, m_bl_no, m_doc_no
122 STORE 0 TO m_o_val, m_comm, m_post, m_ins, m_custom, m_lbr, m_total
123 STORE SPACE(3) TO m_o_knd
124 STORE SPACE(12) TO m_o_des, m_sh_nm, m_co_rm, m_co_ad, m_b_rm, m_b_ad
125 STORE CTOD(" / / ") TO m_o_dat
126 @ 10, 4 SAY "Order #"
127 @ 10, 28 SAY m_ord_no
128 @ 10, 42 SAY "Order value"
129 @ 10, 64 GET m_o_val PICTURE "9,999.99"
130 @ 11, 4 SAY "Order kind(FOB/CAD/CAF)"
131 @ 11, 28 GET m_o_knd
132 @ 11, 42 SAY "Order date"
133 @ 11, 64 GET m_o_dat
134 @ 12, 4 SAY "Order description"
135 @ 12, 28 GET m_o_des
136 @ 12, 42 SAY "Shipping name"
137 @ 12, 64 GET m_sh_nm
138 @ 13, 4 SAY "Company name"
139 @ 13, 28 GET m_co_rm
140 @ 13, 42 SAY "Company address"
141 @ 13, 64 GET m_co_ad
142 @ 14, 4 SAY "Bank name"
143 @ 14, 28 GET m_b_rm
144 @ 14, 42 SAY "Bank address"
145 @ 14, 64 GET m_b_ad
146 @ 15, 4 SAY "Bill of loading #"
147 @ 15, 28 GET m_bl_no
148 @ 15, 42 SAY "Document shipping #"
149 @ 15, 64 GET m_doc_no
150 @ 18, 5 SAY "Commision"
151 @ 18, 17 GET m_comm PICTURE "9,999.99"
152 @ 18, 28 SAY "Insurance"
153 @ 18, 38 GET m_ins PICTURE "9,999.99"
154 @ 18, 51 SAY "Postage"
155 @ 18, 65 GET m_post PICTURE "9,999.99"
156 @ 19, 5 SAY "Labor transportation"
157 @ 19, 38 GET m_lbr PICTURE "9,999.99"
158 @ 19, 51 SAY "Customs"
159 @ 19, 65 GET m_custom PICTURE "9,999.99"
160 @ 20, 5 SAY "Total"
161 @ 9, 2 TO 16, 77
162 @ 17, 2 TO 21, 77
163 CLEAR GETS

```

```

164 RETURN
165
166 *****
167 * Procedure : ordprd
168 * Purpose   : Views order(s) according to specific date(s)
169 *****

170 PROCEDURE ordprd
171 page = 1
172 DO WHILE .NOT. EOF()
173     DO border with scrtitle,scrfootr
174     DO ord_fmt
175     @ 6,65 SAY "Page no."+STR(page,3)
176     @ 10, 28 GET ORD_NO
177     @ 10, 64 GET ORD_VAL PICTURE "9,999.99"
178     @ 11, 28 GET ORD_KIND
179     @ 11, 64 GET ORD_DAT
180     @ 12, 28 GET ORD_DESC
181     @ 12, 64 GET SHIP_NAM
182     @ 13, 28 GET COMP_NAM
183     @ 13, 64 GET COMP_ADR
184     @ 14, 28 GET BNK_NAM
185     @ 14, 64 GET BNK_ADR
186     @ 15, 28 GET BILL_NO
187     @ 15, 64 GET DOC_NO
188     @ 18, 17 GET COMISION PICTURE "9,999.99"
189     @ 18, 38 GET GOOD_INS PICTURE "9,999.99"
190     @ 18, 65 GET POSTAGE PICTURE "9,999.99"
191     @ 19, 58 GET LBR_TRNS PICTURE "9,999.99"
192     @ 19, 65 GET CUSTOM PICTURE "9,999.99"
193     @ 20, 17 SAY TOTAL PICTURE "9,999.99"
194     CLEAR GETS
195     @ 9, 2 TO 16, 77
196     @ 17, 2 TO 21, 77
197     @ 24,2 SAY " Display more records? [Y/N] " GET mor PICTURE "!"
198     READ
199     IF mor = "N"
200         <-----EXIT
201     ENDIF
202     SKIP
203     page=page+1
204     IF EOF()
205         @24,2
206         @24,2 SAY "***** End of data. Press any key *****"
207         READ
208     ENDIF
209 ENDDO
210 RETURN
211
212 *****
213 * Procedure : inv_fmt
214 * Purpose   : Drows item(s) screen format
215 *****

216 PROCEDURE inv_fmt
217 PUBLIC m_item, m_locat, m_desc, m_model, m_on, m_price, m_reorder
218 * Gives those variables their starting values
219 STORE SPACE(5) TO m_item, m_locat
220 STORE SPACE(12) TO m_desc

```

الملحق الثالث : ملف الاجراءات DBPROC.PRG

```

221 STORE SPACE(10) TO m_model
222 STORE 0 TO m_on, m_price, m_reorder
223 @ 10, 6 SAY "Item No."
224 @ 10, 28 224 @ 10, 28 GET m_item
225 @ 12, 6 SAY "Description"
226 @ 12, 28 GET m_desc
227 @ 14, 6 SAY "Model"
228 @ 14, 28 GET m_model
229 @ 16, 6 SAY "Location"
230 @ 16, 28 GET m_locat
231 @ 18, 6 SAY "Quantity on stock"
232 @ 18, 28 GET m_on PICTURE "99999"
233 @ 20, 6 SAY "Unit price"
234 @ 20, 28 GET m_price PICTURE "9,999.99"
235 @ 20, 44 SAY "Reorder point"
236 @ 20, 60 GET m_reorder PICTURE "99999"
237 CLEAR GETS
238 RETURN
239
240 *****
241 * Procedure : bil_fmt
242 * Purpose : Drows bill(s) screen format
243 *****

244 PROCEDURE bil_fmt
245 CLEAR
246 @ 0,5 SAY "Invoice No. "+RTRIM(invno)
247 @ 1,5 SAY "===== "+ REPLICATE(CHR(61),LEN(TRIM(invno)))
248 @ 3,0 SAY REPLICATE(CHR(196),80)
249 @ 4,0 SAY " Item No. Description"+SPACE(9)+"Quantity"+ ;
249 SPACE(6)+"Price"+SPACE(18)+"Amount"
250 @5,0 SAY REPLICATE(CHR(196),80)
251 LIST OFF item_no,SPACE(3),B->desc,SPACE(5),qty_rcv,SPACE(7), ;
251 price,SPACE(12),qty_rcv * price
252 SUM qty_rcv*price TO bttotal
253 @ ROW()+1,64 SAY "-----"
254 @ ROW()+1,40 SAY "Total invoice"
255 @ ROW(),66 SAY bttotal
256 SKIP - 1
257 @ROW()+1,40 SAY "Discount"
258 @ROW(),72 SAY b_dscnt
259 fttotal=bttotal - b_dscnt
260 @ ROW()+1,64 SAY "-----"
261 @ROW()+1,40 SAY "Final total"
262 @ROW(),66 SAY FTOTAL
263 @ ROW()+1,64 SAY "=====
264 RETURN
265
266 *****
267 * Procedure : ps wrd
268 * Purpose :
269 *****

270 PROCEDURE ps wrd
271 n=0
272 @ 22,05 SAY "Enter password (Up to 8 characters) -->"
273 DO WHILE .T.
274 | SET CONSOLE OFF
275 | ACCEPT TO pass

```

```

276 | pass = UPPER(pass)
277 | IF TRIM(pass)= "MAGDI"
278 | SET CONSOLE ON
279 |-----RETURN
280 | ELSE
281 |     n=n+1
282 |     IF n=5
283 |         SET CONSOLE ON
284 |         CLEAR
285 |         WAIT " Sorry ... You are not allowed "
286 |-----RETURN TO MASTER
287 |     ENDIF
288 |     SET CONSOLE ON
289 |     @ 22,5 SAY "Password not matched . Try again....."
290 |-----LOOP
291 | ENDIF
292 ENDDO
293 RETURN
294
295 *****
296 * Procedure : cust_fmt
297 * Purpose : Drows customer screen format
298 *****
299 PROCEDURE cust_fmt
300 * Make the variables globally available
301 PUBLIC m_cust, m_company, m_addr, m_fnam, m_mnam, m_lnam, m_city, ;
301 m_phone, m_bal, m_mxcrdt, m_stbal, m_stdat
302 STORE SPACE (04) TO m_cust
303 STORE SPACE (20) TO m_company, m_addr
304 STORE SPACE (12) TO m_fnam, m_mnam, m_lnam
305 STORE SPACE (17) TO m_city
306 STORE SPACE (08) TO m_phone
307 STORE 0 TO m_bal, m_mxcrdt, m_stbal
308 STORE CTOD(" / / ") TO m_stdat
309 @ 10, 05 SAY "Customer ID. :"
310 @ 10, 20 SAY m_cust
311 @ 11, 05 SAY "Customer name:"
312 @ 11, 20 GET m_fnam
313 @ 11, 34 GET m_mnam
314 @ 11, 47 GET m_lnam
315 @ 12, 05 SAY "Company Name :" GET m_company
316 @ 13, 05 SAY "Address      :" GET m_addr
317 @ 14, 05 SAY "City          :" GET m_city
318 @ 14, 44 SAY "Phone           :" GET m_phone PICTURE "999-9999"
319 @ 17, 05 SAY "Current balance  :" GET m_bal PICTURE "9,999.99"
320 @ 17, 44 SAY "Maximum credit  :" GET m_mxcrdt PICTURE "9,999.99"
321 @ 18, 05 SAY "Last stmt. date  :" GET m_stdat
322 @ 18, 44 SAY "Last stmt. balace:" GET m_stbal PICTURE "9,999.99"
323 CLEAR GETS
324 RETURN
325
326 *****
327 * Procedure : scrol
328 * Purpose : To accept keystrokes from user to scroll the screen *
329 * Parameters : SCNT - Returns vertical scroll amt to next screen *
330 * C_LINE - Number of lines displayed on last screen *
331 * M_SCRL - Number of lines that fit on a screen *
332 *****

```

```

333 PROCEDURE scrol
334 PARAMETERS scnt,c_line,m_scl
335 PRIVATE i
336 sclmsg = "Use " + CHR(24)+" "+CHR(25)+" " + " "+CHR(17)+CHR(196)+ ;
337 CHR(217)+ " Keys, PgUp ,PgDn, Home (BOF), " +"or End (EOF); Esc to quit"
338 * Wait for one of specified keys
338 @ 23, 2 GET sclmsg
339 i=0
340 SET ESCAPE OFF      && Don't interrupt processing on pressing Esc key
341 DO WHILE .not. STR(i,2) $ "13, 5,24,18, 3,27, 1, 6" && Valid keypresses
342 |     i=INKEY()      && Read keypress
343 ENDDO
344 SET ESCAPE ON      && If Esc key pressed, stop processing.
345 * Reset record pointer and cumulative count based on key pressed
346 DO CASE
347 CASE i=27          && ESC key
348 |     * flag to Exit the display loop
349 |     scnt=999
350 CASE i=5           && UP arrow key
351 |     * Move record ppointer to one line above start
352 |     scnt=-(c_line+1)
353 CASE i=24          && DOWN arrow key
354 |     * Move record pointer down one line
355 |     scnt=-(c_line-1)
356 CASE i=18          && PgUp key
357 |     * Scroll up one screen
358 |     scnt=-(c_line+m_scl)
359 CASE i=3 .OR. i=13 && PgDn or ENTER key
360 |     * Scroll down one screen
361 |     scnt=0
362 CASE i=1           && Home key
363 |     * Move to beginning of filtered file
364 |     scnt=0
365 |     GO TOP
366 CASE i=6           && END key
367 |     * Move to end of filtered file
368 |     scnt=-(m_scl-1)
369 |     GO BOTTOM
370 ENDCASE
371 RETURN
372
373 *****
374 * Procedure      : arb_mod
375 * Purpose        : To call DB_API with arabization options
376 *****
377 PROCEDURE arb_mod
378 CALL DB_API WITH 'CODEPAGE ARABIC'
379 CALL DB_API WITH 'NUMER HINDU'
380 CALL DB_API WITH 'SCREEN ARABIC'
381 CALL DB_API WITH 'ACSD ON'
382 CALL DB_API WITH 'DISPLAY VIRTUAL'
383 CALL DB_API WITH 'PRINT_ORIEN ARABIC'
384 CALL DB_API WITH 'PRIN_LANG NO'
385 * ---- Notes:
386 * You may set MA/2 environment by loading a pre-defined macro
387 * with the following syntax:
388 * CALL DB_API WITH 'MACRO <MA macro filename>'
389 * ----

```

```

390
391 *****
392 * Procedure      : lat_mod
393 * Purpose        : To call DB_API to reset arabization options
394 *****

395 PROCEDURE lat_mod
396 CALL DB_API WITH 'NUMER ARABIC'
397 CALL DB_API WITH 'SCREEN LATIN'
398 CALL DB_API WITH 'PRINT_ORIEN LATIN'
399 CALL DB_API WITH 'PRIM_LANG YES'
400 * ---- Notes:
401 * You may set MA/2 environment by loading a pre-defined macro
402 * with the following syntax:
403 * CALL DB_API WITH 'MACRO <MA macro filename>'
404 * ----
405
406 *****
407 * Procedure      : bordarb
408 * Purpose        : Drows a screen frame
409 *****

410 PROCEDURE bordarb
411 PARAMETERS scrtitle,scrfootr
412 PUBLIC systitle,title2,mdate,mtime
413 systitle = "شركة الخشب العربي"
414 title2   = "بي ت : 39387"
415 mdate   = " : التاريخ : "+DTOC(DATE())
416 mtime   = " : الوقت : "+TIME()
417 CLEAR
418 DO intense WITH 0, 1, mdate
419 DO intense WITH 0,78 - LEN(mtime), mtime
420 IF mcolor
421 | SET COLOR TO R+/N
422 ELSE
423 | SET COLOR TO W+/N
424 ENDIF
425 @ 1, 1 TO 4,78 DOUBLE
426 @ 7, 1 TO 23,78 DOUBLE
427 @ 2,80/2-(LEN(systitle)/2) SAY systitle
428 @ 3,80/2-(LEN(title2)/2) SAY title2
429 @ 8,80/2-(LEN(scrtitle)/2) SAY scrtitle
430 @ 9,2 TO 09,77
431 @21,2 TO 21,77
432 @22,5 SAY scrfootr
433 SET COLOR TO W+/N
434 RETURN          && End of procedure
435
436 *****
437 * Procedure      : arb_con
438 * Purpose        : To arabize screen and keyboard
439 *****

440 PROCEDURE arb_con
441 ?CHR(27)+"fa"          && Set arabic screen orientation
442 ?CHR(27)+"fe"          && Set keyboard language to arabic
443 *
444 * ?CHR(27)+"f00"        && Disable screen orientation switching by use
445 * ?CHR(27)+"fK0"        && Disable keyboard language switching by use

```

الملحق الثالث : ملف الاجراءات DBPROC.PRG

```
446 RETURN
447
448 *****
449 * Procedure : lat_con
450 * Purpose   : To reset arabizing screen and keyboard
451 *****

452 PROCEDURE lat_con
453 ?CHR(27)+"f|"      && Set english screen orientation
454 ?CHR(27)+"fk"      && Set keyborad language to latin
455 *
456 * ?CHR(27)+"f01"    && Enable screen orientation switching by use
457 * ?CHR(27)+"fk1"    && Enable keyboard language switching by use
458 RETURN
459
460 *****
461 * End of DBPROC.PRG *
462 *****
```


سلسلة تيسير علوم الحاسب للمؤلف

صدر منها:

- ١ المرجع الأساسي لقاعدة البيانات dBASE III PLUS
- ٢ المرجع الشامل لقاعدة البيانات dBASE III PLUS
- ٣ المرجع الأساسي لقاعدة البيانات dBASE IV
- ٤ المرجع الشامل لقاعدة البيانات dBASE IV
- ٥ المرجع الأساسي لمستخدمي لوتس ١ - ٢ - ٣
- ٦ المرجع الأساسي لنظام تشغيل الحاسبات الشخصية DOS
- ٧ المرجع الأساسي لقاعدة البيانات Clipper
- ٨ التطبيقات التجارية باستخدام dBASE III PLUS

هذا الكتاب

يتخاطب هذا الكتاب المبرمجين الذين يرغبون في تطوير نظم إدارة قواعد البيانات وبيعها في الأسواق. ورجال الأعمال الذين يودون تطوير نظم تجارية لخدمة أعمالهم. ويشتمل الكتاب على التطبيقات التالية:

- نظام الاعتمادات المستندية.
- نظام المشتريات.
- نظام المبيعات.
- نظام المخازن.
- نظام حسابات العملاء.
- نظام صيانة الملفات.

ويمكن تشغيل هذه الأنظمة منفردة أو ربطها معاً. كما يمكن تعديلها حسب رغبة العميل وبيعها في الأسواق. ويوفر الكتاب وقت وجهد البرمجة باحتوائه على ما يقرب من الخمسة آلاف أمراً هي مجموع أوامر البرامج.

